

# AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

*Semanal*

AÑO II N.º 85

190 Ptas.

## INICIACIÓN

Todo sobre  
el disco del CPC  
-Comandos y trucos



### TRUCOS

Búsqueda  
y sustitución  
de palabras  
en programas  
Basic



### INFOBYTES

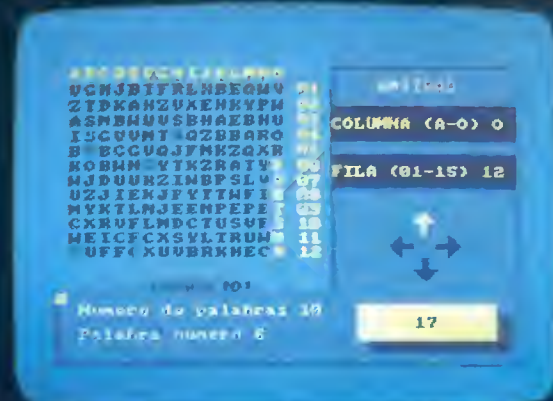
- No más cortes de luz
- Módem para todos los AMSTRAD



### SERIE ORO

Para los más pequeños:  
Sopa de letras

AMSTRAD CPC



### CÓDIGO MÁQUINA

Como rotar cualquier objeto  
en pantalla rápidamente



AMSTRAD CPC

HOBBY PRESS

### JUEGOS

- Palitrón
- Uchimata
- Ace of aces
- Donkey kong
- Explorer





CINCO MINUTOS ANTES DE COMPRAR UN JUEGO A **875 Ptas.**  
 ■ ECHALE UN VISTAZO A ESTOS JUEGOS DE **875 Ptas.**



**875 Ptas.**  
 (Versión Cassette)

**COBRA**  
**Z1**  
 SOFTWARE

**SÍGUENOS EL JUEGO.**



**Director Editorial**  
Joaquín Gómez Centurión

**Director Ejecutivo**

José M.<sup>a</sup> Díaz

**Redactor Jefe**

Juan José Martínez

**Diseño y maquetación**

Rosa María Capitel,

Valeriano Cenalmor

**Redacción**

Eduardo Ruiz de Velasco

y Carmen Elías

**Colaboradores**

Javier Barceló, David Sopena,

Robert Chatwin, Antonio Cuadra,

Pedro Sudón, Miguel Sepúlveda,

Francisco Martín, Jesús Alonso.

Pedro S. Pérez, Amalio Gómez,

Alberto Suñer

**Secretaría Redacción**

Marisa Cogorro

**Fotografía**

Carlos Candel

Chema Sacristán

Miguel Lamana

**Ilustradores**

J. Igual, M. Barco, J. Siemens.

F. L. Frontán, Pejo,

**Edita**

HOBBY PRESS, S.A.

**Presidente**

María Andrino

**Consejero Delegado**

José I. Gómez-Centurión

**Jefe de Producción**

Carlos Peropadre

**Jefe de Publicidad**

Mar Lumberras

**Jefe de Administración**

Raquel Jiménez

**Redacción, Administración  
y Publicidad**

Ctra. de Irún km 12,400

(Fuencarral) 28049 Madrid

**Pedidos y suscripciones:**

734 65 00

Redacción: 734 70 12

**Dto. Circulación**

Paulino Blanco

**Dto. Marketing**

Emilio Juárez

**Distribución**

Coedis, S. A. Valencia, 245

Barcelona

**Imprime**

ROTEDEC, S. A. Crta. de

Irún. Km. 12,450 (MADRID)

**Fotocomposición**

Novocomp, S.A.

Nicolás Morales, 38-40

**Fotomecánica**

GROF

Ezequiel Solana, 16

**Depósito Legal:**

M-28468-1985

Derechos exclusivos  
de la revista

**COMPUTING with  
the AMSTRAD**

Representante para Argentina, Chile,  
Uruguay y Paraguay, Cia. Americana de  
Ediciones, S.R.L. Sud América 1.532. Tel.  
21 24 64. 1209 BUENOS AIRES (Argentina).

AMSTRAD Semanal no se hace  
necesariamente solidaria de las opiniones  
vertidas por sus colaboradores en los artículos  
firmados. Reservados todos los derechos.

# EDITORIAL

**U**na de las principales ventajas de los Amstrad CPC es que tienen una unidad de disco. Ya hemos hablado otras veces de ella, desde distintos puntos de vista, pero nunca hemos analizado tan exhaustivamente todos los comandos de manejo de disco, uno por uno y con tanta claridad. Creemos que, tanto a los que empiezan como a los que deseen aclarar conceptos acerca del disco, les será de gran utilidad.

Continuamos con la segunda parte del artículo de gráficos por ordenador, Ahorrar memoria en gráficos. En él se detallan aún más técnicas que la semana anterior para conseguir empaquetar dibujos en el mínimo espacio posible. Los métodos de ahorro de memoria se ven complementados por unas cuantas rutinas de giro de objetos en distintos ángulos. Como los programas están escritos en lenguaje máquina, no les quiero decir a qué sorprendente velocidad trabajan. Además, y como de costumbre, es muy fácil incluirlas en los programas que desarrollamos.

Los aficionados a «picarse» programas no se quejarán. En este número ofrecemos material lo suficientemente original y adictivo para dejarle al lector el placer de descubrirlo.

Una última recomendación: no se pierdan los trucos. También les sorprenderán.

## SUMARIO

4 Hoy por Hoy

6 Introducción Comandos de disco

11 Sopa Oro Sopa de letras

15 Mundo del CPC  
Complementos para Newton

20 Previas

26 Trucos

30 Gráficos por ordenador

35 Catálogo de Software

40 Mercado común

44 Infobytes.

46 Código máquina

50 Libros.

## Nuevo disco duro para el PCW

La empresa ASD Peripherals ha creado una unidad de disco duro para el Amstrad PCW. Su capacidad es de diez o veinte megabytes y va acompañada de una versión modificada del CP/M Plus, Locoscript y varios programas más de utilidades que aumentan las posibilidades de uso del sistema.

Con todo esto, el PCW puede emplearse como algo más que un simple procesador de textos. Al aumentar su capacidad de almacenamiento puede convertirse en una buena base de datos y en un importante sistema de oficina, teniendo los programas apropiados, ya que



en diez megabytes se pueden almacenar tres mil páginas de texto. Además, el acceso a los datos es muy rápido, concretamente 85 ms.

Como colofón de todas estas aplicaciones, el PCWHD va acompañado de unas claras explicaciones que hacen muy sencilla su utilización, así como



su instalación a través de un simple cable.

## Nota aclaratoria

En el número 77 (página 44), «Especial Ampliaciones» de la sección de Infobytes, se decía que el periférico *Anta 64*, de MHT Ingenieros, convierte un Amstrad CPC 664 en un 6128

Dicha información no es del todo exacta. La ampliación de memoria no proporciona al 664 los comandos y el Sistema Operativo ampliado que caracterizan al 6128.

El *Anta 64K.3* es una ampliación de memoria de 64 K con tres opciones distintas:

- Buffer de impresora, que almacena y envía la información a la impresora, dejando el ordenador libre para otros trabajos.

- Ampliación de memoria, en la que se pueden leer datos, grabarlos, etc.

- Ram-Disk, el cual se comporta como un disco RAM de acceso muy rápido, incluyendo funciones para Acceso Aleatorio.

La única limitación del *Anta* viene dada por las propias características de este aparato. En cuanto se apague el ordenador los datos que se han incluido en la ampliación Ram de la memoria se perderán.

## Zafiro se suma a la reducción de precios

Una excelente noticia para los usuarios de ordenadores domésticos: los responsables de la sección de software de la compañía Zafiro han aplicado una considerable reducción en el precio de la totalidad de los programas de su catálogo, con lo que dicha casa de distribución se suma a las últimas tendencias generalizadas del mercado, y viene a ampliar la gama de programas a precios asequibles.

Según han manifestado los propios directivos de Zafiro, esta idea de la baja venía fraguándose durante algún

tiempo en la compañía, y el hecho de que otras casas se adelantaran en la puesta en marcha de la disminución de los precios, les ha obligado a modificar ligeramente su estrategia y a acelerar su realización.

De esta forma, las medidas tomadas por Zafiro pasan por la creación de un nuevo sello de software, *Cobra*, y por la conservación de la tradicional firma Zafichip, aunque con precios más bajos y asequibles para sus títulos.

En el sello *Cobra* se incluirán la totalidad de los grandes éxitos que Zafichip ha presentado hasta el momento («Commando», «Trap Door», «1942», «Nosferatun», etc...), títulos a los que se irán sumando las próximas

novedades de compañías tan prestigiosas como *Piranha* o *Elite*. El PVP de todos estos programas será de 875 pesetas.

Por otra parte, Zafiro ha manifestado la intención de que, por razones de prestigio, el sello *Zafichip* siga existiendo, por lo que en él se incluirán los programas que, a juicio de la compañía, posean una calidad superior a la del resto, y que estén llamados a convertirse en superéxitos. Este tipo de programas también verá reducido considerablemente su precio, el cual quedará establecido en la cifra de 1.200 pesetas.

Afortunadamente, cada vez se nos ponen mejor las cosas a los usuarios y el software continúa haciéndose más asequible para todos.



## Atari relanza su gama de productos ST

El miércoles 8 de abril, *Ordenadores Atari, S.A.*, a la que podríamos llamar Atari España para entendernos, presentó, o más bien se podría decir representó, su gama de productos ST. Se trata de unas máquinas muy prometedoras, basadas en uno de los microprocesadores más avanzados del mercado: el *Motorola 68000* corriendo, nunca mejor dicho, a 8 MHz. Como atractivos adicionales, los *ST512* y *ST1040* incorporan medio mega y un mega de memoria, respectivamente, y un entorno de usuario basado en el Gem, lo mismo en el *Amstrad PC*, pero con la ventaja a favor de Atari de una velocidad mucho mayor, y la desventaja de que el Gem se encuentra en Rom, por lo que incluir posteriores versiones de este entorno gráfico de trabajo podría ser problemático. Por lo demás, pensamos que estos ordenadores, para las prestaciones que tienen, se ofrecen a un precio por demás competitivo. Por ejemplo, el *ST512*, sin monitor, pero con teclado, unidad central, ratón y software, cuesta 80.000 ptas.

La representación, para ser justos, también dio algo nuevo e interesante por parte de Atari. Nos referimos a la nueva serie de máquinas *Mega 2* y *Mega 4*, que, realmente, son casi, casi como los otros, excepto por el hecho de que poseen más memoria, 2 y 4 megas, y por un aspecto mucho más modular y profesional. No obstante, la inclusión en la serie *Mega* de un chip especialmente dedicado al proceso de gráficos, lo convierte en adecuados candidatos a formar parte de la caterva de sistemas dedicados al arte o al CAD. Además, en este tipo de aplicaciones es esencial que el software sea «amistoso», ámbito donde el ratón y el Gem tienen mucho que decir. De momento, no hay noticias oficiales ni definitivas acerca de los precios de estas nuevas máquinas. Atari España nos ha facilitado una lista del software ya disponible

## Estadística para el PC

*MicroM* se ha comercializado una nueva herramienta de trabajo para el programa de estadística *SPSSx*. Con ella es posible realizar automáticamente las típicas funciones de gestión de sistema que se realizan con dicho programa, sin necesidad de tener experiencia ni conocimientos especiales del *SPSSx*.

Como todo lo relacionado con el *SPSSx*, el *SPSSx Track* está indicado para los compatibles de IBM, entre ellos el *Amstrad PC*.

Entre las tareas concretas que realiza se encuentran la posibilidad de generar informes diarios, semanales o

mensuales sobre el funcionamiento del sistema, con un gráfico, como ejemplo, de los fallos de acceso al disco.

Otra de sus funciones es documentar los problemas de explotación, así como planificar las ampliaciones del equipo de una manera coherente.

En cuanto a las tareas de gestión diarias, ayuda a automatizar rutinas como detección de anomalías en el sistema de seguridad, informes sobre tamaño y antigüedad de los ficheros, asignación de prioridades y cualquier función diseñada por el responsable del sistema.



para sus ordenadores, y, entre eso y lo que hemos visto, se puede decir que estas máquinas están arropadas por una gran

cantidad de buen software, sin llegar, evidentemente, al número de programas de que dispone IBM.

# Ordenar los ficheros

Por: David Sopena

*En un disco magnético se pueden guardar un montón de cosas, aunque parezcan tan pequeños como son los del Amstrad CPC. Toda esta información necesita ser tratada convenientemente para no perder ni un sólo bit y, además, estar ordenada de una manera tal que no resulte difícil a nuestro ordenador acceder a ella.*

El Amstrad es, solamente, una encantadora máquina que se limita a obedecer fielmente, eso sí, todas las órdenes que le demos, siempre que estén escritas correctamente y pueda interpretarlas sin error. Queremos decir con esto que vamos a ser nosotros los que ordenaremos y trataremos realmente la información contenida en un disco.

En algún que otro artículo anterior vimos poco a poco una de las formas de pasar programas desde la memoria central del ordenador a la unidad de disco o cinta, o en sentido contrario: SAVE y LOAD se encargaban de hacerlo.

Ahora bien, ¿qué tipo de información era la que trasvasábamos de un sitio a otro? Hay que admitir que estábamos bastante limitados. Sólo podíamos pasar dos tipos de ficheros: programas escritos en Basic o contenidos binarios de las distintas direcciones comprendidas en una determinada zona de memoria. ¿Cómo logrará el ordenador acceder a ella correctamente? Dentro del disco hay unos sectores reservados para almacenar ciertos datos de todos los ficheros que contiene: nombre y apellidos, dirección, etc. No vayamos a pensar con esto que un disco es una agenda dispuesta a guardar toda la filiación





de nuestros amiguetes, ¡no! Cuando hablamos de nombre y apellidos de ficheros nos estamos refiriendo concretamente a su «nombre» y a su «extensión».

Lo de «nombre» lo tenemos más o menos claro, pero ¿qué es la «extensión» de un fichero de disco?

Esos sectores reservados para almacenar esta información están preparados para guardar, además del nombre, tres caracteres más que nos van a dar idea de lo que hay en realidad en un determinado fichero. ¿Nos explicamos? Si encontramos algo llamado:

#### NOMBRE.BAS

es inmediato pensar que estamos ante un programa escrito en Basic. «NOMBRE» es el nombre del fichero y «BAS» son los tres caracteres de la «extensión» a los que antes nos referíamos. Estamos seguros que no os parecerá muy descabellado asignar «BAS» a todos los ficheros Basic.

Extendamos esto a cualquiera, incluso a los que de momento nos conocemos su significado. A todos los que contengan datos les colocaremos la extensión o «apellido» DAT, los de texto llevarán TXT, los binarios BIN, las pantallas SCR, y así hasta que se agote nuestra imaginación. Solamente os daremos un consejo: las tres letras os indicarán «algo», así que elegidlas

## INICIACIÓN

adecuadamente para que siempre os recuerden la naturaleza del fichero.

La mayoría de nosotros ya conocemos suficientemente el trabajo de un comando generalmente muy utilizado: CAT. Visualiza los nombres y las respectivas extensiones de todos los ficheros guardados en cualquier disco y además nos informa también del espacio del mismo que queda libre.

Muy bonito, pero con un pequeño inconveniente: la información que nos presenta en la pantalla es la referente a todos, repetimos «todos», programas, pantallas, textos, etc., los contenidos del disco. Eso sí, ordenados alfabéticamente, pero juntos y revueltos.

Nuestra intención es, por ejemplo, conocer los programas Basic almacenados (los de la extensión BAS). ¿Cómo lograrlo?

El **Amstrad** queda sometido al control de un sistema operativo de manejo de disco, llamado AMSDOS, desde el mismo momento que lo encendemos. Algunas de sus funciones ya las hemos visto (LOAD, RUN, etc.) pero nos quedan todavía muchas por descubrir.

El manejo de los discos lo haremos por medio de unos comandos AMSDOS que siempre escribiremos precedidos del símbolo «I»: son las llamadas órdenes externas. Y una de ellas es la que se encargará de solucionar nuestros problemas.

Teclead:

#### IDIR

seguido de RETURN y al momento os aparecerán en la pantalla todos los ficheros que vimos con el comando CAT. Pero esta vez lo han hecho de una manera diferente, así que no podemos pensar que se trata de dos órdenes equivalentes.

Démosle mucha más potencia. Existen unos símbolos llamados «comodines» que nos van a servir para decirle al **Amstrad** que alguna de las órdenes del AMSDOS actúe sobre un conjunto de ficheros que tengan algo en común. Por ejemplo, queremos sacar el «directorio» de todos los ficheros con extensión BAS tal como dijimos antes: esto quiere decir que al comando sólo le interesa la extensión, o sea que el nombre le ha de ser «indiferente».

Para indicárselo al ordenador utilizaremos el símbolo asterisco (\*), en sustitución de todos y cada uno de los nombres de ficheros y el efecto será el deseado.

¿Cómo se hace esto en la práctica? Escribid:

IDIR, «\*».BAS»

y observaréis la diferencia de lo que nos aparece





ahora en la pantalla con respecto a lo anterior: sólo saca los ficheros que tienen BAS por extensión.

Si queremos conocer el «directorio» de todos los que sean binarios haremos tres cuartos de lo mismo:

IDIR,“\*”.BIN”

Y así con cualquier extensión. Comprobadlo. El símbolo comodín asterisco sirve para más cosas. Ahora se nos ha antojado ver todos los programas que se llamen “PEPITO”, por ejemplo. Da igual la extensión que tengan, sólo queremos conocer los que se llamen “PEPITO”.

En esta ocasión, lo que ha de ser indiferente para el Amstrad es el tipo o extensión: da igual. Así que lo sustituiremos por el símbolo comodín que conocemos y a ver qué pasa. Con:

IDIR,“PEPITO.\*”

lo comprobaremos.

¿Existe alguno? Si no es así dadle otro nombre de los muchos, esperamos que tengáis en vuestro disco y veréis que los resultados son totalmente satisfactorios.

Nos atendremos a haceros una pregunta. A qué es equivalente la orden:

IDIR,“\*.\*”

Con ella estamos indicándole al ordenador, o mejor al sistema operativo del manejo de disco, que nos de un directorio de ¿qué? Analicémoslo. Decimos que queremos cualquier nombre de fichero (así nos lo indica el primer asterisco) y además nos es indiferente la extensión que tenga (segundo asterisco). O sea, le pedimos todos los contenidos en el disco, lo mismo que hacíamos con IDIR.

De golpe tenemos la inquietud de saber si existe alguno, cuyo nombre empiece por la letra D. Conociendo que “\*” sustituye a un grupo de caracteres que no os costará mucho trabajo comprender que:

IDIR,“D\*.\*”

nos dará lo que pedimos.

Existen más símbolos comodines. La

interrogación “?” cumple la misma misión que el asterisco pero en este caso con un solo carácter.

IDIR,“PEP?.\*”

visulizará el directorio de todos los ficheros, de cualquier tipo, cuyo nombre contenga cuatro caracteres y que además comiencen por “PEP”, siendo el cuarto indiferente.

Serían equivalentes las órdenes:

IDIR,“\*.\*”

y

IDIR,“?????????.”

Estamos seguros que vuestra respuesta habrá sido afirmativa ya que en ambos casos nos estamos refiriendo a «todos los ficheros».

¿Ampliamos los comandos AMSDOS?

Normalmente es bastante difícil que un programa nos salga bien a la primera y conforme vamos corrigiendo comprobaremos que se nos llena el disco de ficheros con extensión BAK. ¿Qué quiere decir esto?

Sencillamente que el AMSDOS asigna automáticamente dicha extensión a la versión anterior de un fichero cuando hemos grabado una más moderna con el mismo nombre. Así se nos permite utilizar una versión anterior cuando lo juzguemos oportuno.

Si el programa ya funciona correctamente, ¿para qué tener ocupando espacio en el disco todas estas ediciones antiguas de programas? ¿No es así?

¿Intengamos borrar las que no nos sirvan para nada? Supongamos que nos sobre el fichero “PEPE.BAK”. La manera de hacerlo desaparecer es empleando otra orden externa del AMSDOS:

IERA,“PEPE.BAK”

ERA proviene de la palabra inglesa «erase» (borrar), con lo que su emisión queda bastante clara, ¿no? O sea, borra todos los ficheros que estén de acuerdo con la cadena de caracteres que esté entre comillas.

Queremos decir con esto que también puede admitir símbolos comodines de forma semejante a como lo hacía IDIR.

IERA,“PEPE.\*”

borrará del directorio del disco todos los ficheros que se llamen “PEPE” independientemente del tipo que tengan, y

IERA,“\*.\*BAK”

hará lo propio con todos aquellos que sean una versión antigua de cualquier otro, ya que tendrán la extensión BAK.

Por si se os pasado por la mente os recomendamos que no se os ocurra teclear:

IERA,“\*.\*”

a menos de estar muy seguros de lo que queréis hacer. Con este título comando se os quedará el



disco más blanco que la nieve.

Antes de continuar haremos una puntualización. En realidad no se borran los ficheros del disco, a menos que grabemos otro encima. Lo que ocurre es que en el lugar del directorio reservado para el nombre de los ficheros existentes se pone una indicación delante del mismo y el ordenador lo interpreta como que ya no nos interesa.

Bien, ya sabemos la menra de eliminar de un disco todo aquello que no nos interesa. Pongámonos en otro supuesto. Un programa de contabilidad, por ejemplo, genera unos datos que se almacenan con el nombre "DATOS.DAT". Terminamos el año en curso y comenzamos el siguiente, así que queremos que los datos referentes al año 87 estén contenidos en un fichero cuyo nombre refleje por algún sitio este 87: se impone cambiar de nombre a "DATOS.DAT" y ¿qué mejor nombre que "DATOS87.DAT"?

O sea, que tenemos que «renombrarlo» o «renamearlo». Busquemos en el saco de los nuevos comandos a ver que encontramos. ¡Vaya!, hay algo parecido a RENAME. Si,

**IREN,"NUEVO.EXT","VIEJO.EXT"**

es una orden externa de AMSDOS que da un nuevo nombre o «rebautiza» a un fichero ya existente en el disco.

En nuestro ejemplo tendríamos que escribir:

**IREN,"DATOS87.DAT","DATOS.DAT"**

para almacenar los datos del año 87 con un nombre que sea bastante representativo de lo que allí se encuentra.

¿Qué pasa si ya existe otro fichero con el nombre "DATOS87.DAT". Es evidente que en el disco no pueden existir dos con el mismo nombre y con la misma extensión. En este caso "SI" nos interesa, en principio, guardar el primer fichero ya que puede contener datos válidos.

El AMSDOS ha pensado en ello y no lo hace desaparecer, sino que sencillamente da un mensaje de error advirtiéndonoslo: **DATOS87.DAT already exist** evitando así la desaparición de algún fichero que nos valga, en caso de confundirnos al escribir el nombre.

Se nos ocurre también pensar la manera de cambiar de nombre a un grupo de ficheros: estamos pensando en los símbolos comodín. Pues bien, en esta ocasión no podemos emplearlos, así de claro. El cambio de nombre ha de hacerse de uno en uno, si no intentad:

**IREN,"\*87.DAT","\*.DAT"**

se nos traducirá en el consiguiente mensaje de error:

**Bad command**

¡No intentemos hacer cosas que no están permitidas!

# INICIACIÓN



El directorio del disco está dividido en 16 áreas diferentes (0 a 15) llamadas «áreas del usuario» sobre las que actúan independientemente las distintas órdenes de manejo de disco.

El área implícita, o sobre la que trabajamos nada más encender el ordenador, es el área 0. Si ahora mismo pedimos un CAT nos sacará todos los ficheros que estén en esta zona de usuario (la 0). Nosotros podemos trabajar indistintamente en una cualquiera con tan sólo teclear:

**IUSER,Número de área**

y ya estaremos colocados en ella. Por ejemplo:

**IUSER,3**

nos habrá situado en la zona 3. ¿Qué hay en ella? Visualizadlo mediante:

**IDIR**

por ejemplo. ¿Qué aparece?

Lógicamente que no hay ningún fichero en este área ya que hasta ahora no hemos trabajado en ella. Escribamos un corto programa en Basic:

**10 CLS**

**20 PRINT "ÁREA 3"**

y salvémoslo. Una vez hecha esta operación veamos de nuevo el directorio mediante **IDIR** o **CAT**.

Esta vez aparecerá únicamente este último programa ya que estamos en el área 3 y cualquier orden AMSDOS sólo actúa sobre sus ficheros.

Regresad a la zona 0 usando:

**IUSER,0**

# INICIACIÓN



y comprobad todo lo que os decimos. Como siempre, os invitamos a que realicéis cuantas pruebas se os vayan ocurriendo. Seguimos pensando que esta es la mejor forma de comprobar sobre vuestras «mismas carnes» el funcionamiento de un montón de cosas y sobre todo comprobar una serie de particularidades que sin imaginación no llegaríamos a conocer.

Otra posibilidad que nos ofrece AMSDOS es la de transferir un fichero de un área a otra. Si no queremos que cambie de nombre bastará con usar conjuntamente la orden **IREN**, con todo esto que os hemos contado sobre las áreas de usuario, tecleando:

**IREN,"0:PEPE.BAS","3:PEPE.BAS"**

si así habíamos llamado al programita anterior. Habrá desaparecido de la zona 3, como se puede comprobar con:

**!USER,3**

y

**CAT**

para colocarse en la 0. Aseguraos, ¡hombre!:

**!USER,0**

seguido de:

**CAT**

¿Queréis también cambiarlo de nombre? Pues indicádselo al **Amstrad**.

**IREN,"3:JOSE.BAS","0:PEPE.BAS"**

dará al fichero **PEPE.BAS** de la zona 0 el nombre **JOSE.BAS** y lo asigna al área 3 de nuevo. Esto de los **USER** es una forma de dividir y preparar el disco para ser utilizado por varios usuarios.

Bueno, ¿qué os parece si lo dejamos por esta ocasión? Os invitamos a que nos acompañéis en sucesivas semanas en las que continuaremos contandoos nuevas cosas sobre manejo de disco.

## *Correo..., más rápido...*



Con el fin de acelerar lo más posible el correo, y poder resolver o contestar a todas las dudas y sugerencias que llegan a nuestra redacción, a partir de esta semana os rogamos, en beneficio de todos, consignar en el sobre, en lugar bien visible, una de las denominaciones siguientes:

- **Suscripciones AMSTRAD.** Para todos aquellos casos relacionados con petición de cintas, números atrasados, formalización de suscripciones, devoluciones, etc...
- **Mercado Común AMSTRAD.** Compras, ventas, intercambios, clubs...
- **Serie Oro AMSTRAD.** Para los programas que nos enviéis para su publicación.
- **Sugerencias AMSTRAD.** Para vuestras críticas, sugerencias o cualquier opinión que queráis vertir sobre la revista.



del Basket

DA  
LA  
TALLA

Pídela todas las semanas, porque Gigantes da la talla.



**E**sta sección está dedicada a todas las compras, ventas, clubs de usuarios de **Amstrad**, programadores y, en general, cualquier clase de anuncio que pueda servir de utilidad a los lectores. Todo aquel que lo desee puede enviarnos su anuncio, mecanografiado, a: **HOBBY PRESS, S.A. AMSTRAD Semanal. Apartado de Correos 232 Alcobendas. Madrid.**  
**¡ABSTENERSE PIRATAS!**

## Mercado COMÚN

.....  
**Desearía** contactar con usuarios de **Amstrad 464 CPC** para intercambiar juegos. Tengo alrededor de 50 juegos buenos.

Los interesados escribir o llamar a **César Alonso Pena**. Pl. Sagrado Corazón de Jesús, 2. Tel. (91) 411 72 01 (entre las 5 y las 10). 28002 Madrid.

.....  
**Vendo Amstrad CPC 464** monitor color con manuales, más de 30 revistas, cintas con juegos y utilidades (*Batman, Cyrus II Chess, Bomb Jack, Light*

*Force, Strike Force Harrier*, ensamblador, desensamblador, procesador de textos, copiadores, etc.), joystick, tapadera teclado, libros. Todo por sólo 75.000 ptas. Llamar a **Manolo**. Tel. (91) 255 89 01. Madrid.

.....  
**Para PCW 8256/8512**, por fin en España, Software Public Domain para PCW. Lenguajes (C, Pascal, Forth, Prolog, Lisp), y utilidades (también para CPC 6128). *Aliscalc*: hoja de cálculo con sistema de autodiagnóstico de errores de fórmulas. *Alisbase*: base de

datos, acceso indexial sistema B + Tree, rapidísimo, hasta 32767 registros. *Alisútil*: paquete de utilidades para tratamiento de ficheros. Compresor: ahorra un 40 por 100 de espacio de disco, cuentapalabras. Cifrador de textos, *Alissort*: ordena ficheros. Cada disco sólo 1.750 ptas. **R. Uphoff**. Castillo, 7. 18184 Beas (Granada).

.....  
**Club de Usuarios Amstrad Manresa** compra, vende y cambia programas para CPC 664-6128. Poseemos más de 300

# De chip a chip

**"Sábado Chip", de 17 a 19 h.**



programas entre juegos y utilidades. Últimas novedades en juegos y utilidades. Interesados escribir a *Jordi Mominó i de la Iglesia*. Ctra. Santpedor, 80, 4.º, 2.ª. Tel. (93) 873 02 76. 08240 Manresa (Barcelona).

**Poseo** un PC 1512 y la impresora *Seikosha 1000.I*. Me gustaría intercambiar programas e información con usuarios de Compatibles IBM.

Tengo entre otros: *Flight Simulator, C, Printmaster, Turbo Pascal, Pinball*, etc. Llamar o escribir a *Miguel García Vázquez*. Guadalete. 14, 3. 11012 Cádiz. Tel. (956) 28 56 69 (de 2 a 3).

**Intercambio** programas y fotocopias de libros referentes al PCW, solamente en Málaga, preguntar por *Eugenio* en el tel. 43 53 29.

**Traslado** ficheros generados por Amsfile a ficheros gestionables por Dbase II, a partir de los archivos "Nombre".DAT; "Nombre".DEF; "Nombre".NDA. Interesados escribir a *José Luis Aguado*. Curtidores, 3, 5.º. 47006 Valladolid.

**Vendo Amstrad CPC 6128**, con monitor en fósforo verde, en perfectas condiciones y acompañado de muchos discos con programas de juegos y utilidades. Interesados llamar al (988) 74 66 48 a partir de las 15,30 horas los días laborales. *César*.

**Urge** conseguir los manuales de *Dbase II* y *Pascal Compas*, así como los programas *MS Cobol* y

*Wordstar* con manuales de instrucciones. Los cambio por otros programas y manuales, lista con más de 200. Escribir a *Fermín García Nieto*. Gran Vía, 10, 1-A. 30004 Murcia.

**Deseo** contactar con usuarios del Amstrad PCW para intercambio de programas. Interesados escribir a *Alberto González*. Sueca, 17, pta. 27. 46006 Valencia.

**Vendo Amstrad CPC 6128** FV, cable para conexión de un cassette. 25 discos con los mejores programas del mercado, casi todo utilidades. 9 cintas con programas, 6 cintas vírgenes, las últimas 12 revistas de Amstrad User, y la colección, casi completa, de la revista **AMSTRAD Semanal**, incluida la suscripción hasta

# ip Pestilo Cope

Todos los sábados, de 5 a 7 de la tarde, en "Sábado Chip". Dirigido por Antonio Rua. Presentado por José Luis Arriaza, hecho una computadora. Dedicado en cuerpo y alma al ordenador, y a la informática. Haciendo radio chip... estilo Cope.



**Cadena Cope**

RADIO POPULAR

... de chip a chip







# Complemento a Newton

Por: César Lobato

Mundo del  
CPC

*El artículo que sigue está pensado para que los estudiantes, bien de COU o de los primeros cursos universitarios de una facultad de ciencias, puedan aprovechar la capacidad de cálculo de un CPC en la resolución de un tradicional problema de física muy tedioso de enfrentar a mano. Por ello, no se han hecho muchas concesiones en el sentido de divulgar el lenguaje en el que se explica lo que hace el programa y cómo. Aquellos que estén interesados en el tema lo captarán perfectamente a la primera ojeada.*

**E**ste programa ha sido diseñado como complemento a Newton para tratar los casos con mayor número de masas que Newton no consideraba, pero si se utilizan más masas, su método de ecuaciones resulta demasiado largo y engorroso, por lo que obliga a trabajar con una mecánica de mayor nivel como es la mecánica de Lagrange.

No es que las mecánicas sean diferentes, sino

que la mecánica de Lagrange es más fácil si sabemos derivar.

En la mecánica de Lagrange en vez de utilizar un sistema absoluto de coordenadas, utilizamos una serie de restricciones y de coordenadas relativas al problema. Será en estas coordenadas en las que halleemos el *lagrangiano*, que no es más que la resta de la energía cinética en las nuevas coordenadas y del potencial.

```
10 REM *****
20 REM *** programa hecho ***
30 REM ***
40 REM *** by ***
50 REM ***
60 REM *** CESAR LOBATO ***
70 REM *****
80 REM
90 REM Este programa calcula las ac-
   lera- ciones de masas por lagrang
   e
100 DIM rez(5):ERASE rez
101 DIM masa(5),ang(5),ace(5,5),mat
   riz(4,4),acele(4),ka(5,3)
102 ERASE masa,ang,ace,matriz,acele
   ,ka
110 DIM a(5),b(5,2),pol(4,8),c(4,2)
120 ERASE pol,a,b
130 MODE 1:CLS:DEG:WINDOW#4,1,40,22
   ,25:PAPER#4,3:CLS#4
140 WINDOW#2,1,40,1,5:PAPER#2,3:CLS
   #2
150 b(1,1)=302:b(2,1)=192:b(3,1)=15
   6:b(4,1)=102:b(5,1)=80
160 SOUND 1,239,50,8:PRINT#2,SPC(17
   );"HOLA";SPC(17);SPC(8);"Vamos a re
   solver un problema";SPC(9);SPC(16);
   "de";SPC(18);SPC(11);"planos inclin
   ados";SPC(12);SPC(12);"maquina de A
   TWOOD"
170 PRINT#4,"masas maximas 5":INPUT
   #4,"cuantas masas por favor";n1
180 IF FIX(n1)<n1 THEN CLS#4:GOTO
   170
190 IF n1<=1 OR n1>5 THEN CLS#4:GOT
   O 170
200 d$=CHR$(143)+CHR$(143)
210 FOR i=1 TO n1
220 a(i)=b(n1,1)*i+32*(i-1)
230 IF n1=4 THEN a(i)=a(i)+1
240 MOVE a(i),9*16:TAG:PRINT d$;:TA
   GOFF
250 MOVE a(i),8*16:TAG:PRINT d$;:TA
   GOFF
260 MOVE a(i),6*16:TAG:PRINT "M";i;:
   TAGOFF
```

```
270 NEXT i
280 n4=0:CLS#2:CLS#4
290 na=1:GOSUB 2880
300 REM 9800 mira donde poner el cu
   rsor
310 SOUND 1,150,50,6:PRINT#4,"Princ
   ipio de cuerda ";CHR$(224)
320 n2=na
330 IF n2>n1 THEN m1=pol(n2-n1,6):m
   2=pol(n2-n1,7):f$=CHR$(224) ELSE m1
   =a(n2):m2=9*16:f$=CHR$(224)
340 MOVE m1,m2:TAG:PRINT f$;:TAGOFF
350 PRINT#2,SPC(10);"C Cambiar de o
   pcion";SPC(11);SPC(10);"D Borrar";S
   PC(21);SPC(10);"O eliges Opcion";
360 REM 8000 mira si pulsas O,D o C
370 GOSUB 2680
380 IF p1=2 THEN GOTO 460 ELSE IF p
   1=3 THEN GOTO 530
390 na=n2+1:GOSUB 2880:nb=na
400 IF nb>n1 THEN m3=pol(nb-n1,6):m
   4=pol(nb-n1,7):f2$=CHR$(224) ELSE m
   3=a(nb):m4=9*16:f2$=CHR$(224)
410 IF n2>n1 THEN f1$=CHR$(231):m1=
   pol(n2-n1,6):m2=pol(n2-n1,7) ELSE f
   1$=CHR$(143):m1=a(n2):m2=9*16
420 REM 8100 cambia el cursor de si
   tio
430 GOSUB 2710
440 n2=na:SOUND 1,239,25,7:GOTO 370
450 REM por si no hay polea que bor
   rar
460 IF n4=0 THEN SOUND 1,190,25,7:S
   OUND 1,280,25,7:SOUND 1,350,25,7:LO
   CATE#4,1,2:PRINT#4,"NO HAY QUE BORR
   AR";:GOTO 370
470 ana1=c(n4,1): ana2=c(n4,2)
480 FOR i=1 TO 2:c(n4,i)=0:NEXT i:F
   OR j=1 TO 8:pol(n4,j)=0:NEXT j
490 IF ana1>n1 THEN pol(ana1-n1,8)=
   0 ELSE b(ana1,2)=0
500 IF ana2>n1 THEN pol(ana2-n1,8)=
   0 ELSE b(ana2,2)=0
510 n4=n4-1:CLS#4:PRINT#4,"BORRADA"
   :GOSUB 2730:GOTO 290
```

```
520 REM borro la polea y vuelvo a e
   legir
530 IF n2>n1 THEN pol(n2-n1,8)=1 EL
   SE b(n2,2)=1
540 IF n2>n1 THEN m1=pol(n2-n1,6):m
   2=pol(n2-n1,7):f1$=CHR$(231) ELSE m
   1=a(n2):m2=9*16:f1$=CHR$(143)
550 MOVE m1,m2:TAG:PRINT f1$;:TAGOF
   F
560 na=1:GOSUB 2880
570 REM miro dode poner el cursor
580 SOUND 1,150,50,6:PRINT#4,"FINAL
   de cuerda ";CHR$(225);
590 n3=na
600 f1$=CHR$(225):IF n3>n1 THEN m1=
   pol(n3-n1,6):m2=pol(n3-n1,7) ELSE m
   1=a(n3):m2=9*16
610 MOVE m1,m2:TAG:PRINT f1$;:TAGOF
   F:CLS#2
620 PRINT#2,SPC(10);"C Cambiar de o
   pcion";SPC(11);SPC(10);"D Borrar";S
   PC(21);SPC(10);"O eliges Opcion";
630 REM 8000 mira si pulsas O,D o C
640 GOSUB 2680
650 IF p1=2 THEN GOTO 710 ELSE IF p
   1=3 THEN GOTO 750
660 na=n3+1:GOSUB 2880
670 IF na>n1 THEN m3=pol(na-n1,6):m
   4=pol(na-n1,7):f2$=CHR$(225) ELSE m
   3=a(na):m4=9*16:f2$=CHR$(225)
680 IF n3>n1 THEN f1$=CHR$(231):m1=
   pol(n3-n1,6):m2=pol(n3-n1,7) ELSE m
   1=a(n3):m2=9*16:f1$=CHR$(143)
690 REM cambio el cursor de sitio
700 GOSUB 2710:SOUND 1,239,25,7:n3=
   na:GOTO 640
710 ana1=c(n4,1): ana2=c(n4,2)
720 REM borro la entrada anterior y
   devuel el control a 140
730 IF n2>n1 THEN pol(n2-n1,8)=0 EL
   SE b(n2,2)=0
740 CLS#4:SOUND 1,219,25,6:PRINT#4,
   "BORRADA":GOSUB 2730:GOTO 290
750 IF n3>n1 THEN pol(n3-n1,8)=1 EL
   SE b(n3,2)=1
```

A continuación procedemos a hallar un sistema de ecuaciones donde las variables del sistema serán las aceleraciones relativas, es decir, las segundas derivadas de las coordenadas relativas al problema. Muchísimas veces lo que representan esas ecuaciones son sistemas de ecuaciones

diferenciales que, incluso, sólo pueden ser resolubles por métodos aproximativos, sin llegar nunca a conocer la solución exacta.

Por lo dicho hasta ahora, puede pensarse que el programa es de difícil comprensión; sin embargo, en el caso particular que nos ocupa, el sistema es lineal y con sólo diagonalizar la matriz del sistema obtendremos las aceleraciones relativas.

## SUBROUTINAS

|      |   |
|------|---|
| 1100 | Crea y resuelve el sistema de ecuaciones obtenido del problema mediante el lagrangiano y las ecuaciones de Lagrange.                        |
| 1630 | Presenta un dibujo que es el problema que queremos resolver mostrando masas, pesos, y coordenadas generalizadas.                            |
| 2190 | Presentación de resultados, aceleraciones absolutas (Newton) y relativas (Lagrange). Las absolutas son obtenidas a partir de las relativas. |
| 2360 | Asigna las posiciones de la polea y construye los vectores necesarios para pasar de coordenadas absolutas a relativas.                      |
| 2590 | Control de fin.   |
| 2680 | Mira si pulsas O, D o C.  |
| 2710 | Cambia el cursor de sitio.  |
| 2730 | Dibuja masas y poleas mientras se están metiendo poleas.  |
| 2880 | Mira dónde poner el cursor.   |
| 3000 | Presentación del sistema.   |
| 3500 | Cabecera.   |

## VARIABLES

|        |  |
|--------|--|
| REZ    | Aceleraciones absolutas.   |
| N1     | Número de masas.   |
| MASA   | Guarda el valor numérico de las masas.   |
| ANG    | Guarda el valor de los ángulos en grados.  |
| ACE    | Nos dice las poleas que se ven afectadas por una masa.                                       |
| MATRIZ | Del sistema de ecuaciones de Lagrange.   |
| ACELE  | Aceleraciones relativas.   |
| A      | Posiciones donde poner las masas.  |
| B      | Posiciones donde poner la primera masa según el número de masas y masas que ya están unidas. |
| POL    | Posiciones de poleas y masas que une.  |
| FUX    | Fuerza eje X.  |
| FUY    | Fuerza eje Y.  |
| FUE    | Fuerza paralela al movimiento.   |

```

760 REM asigno las posiciones de 1
a polea y construyo los vectores pa
ra poder aplicar LAGRANGE
770 GOSUB 2360
780 n4=n4+1
790 GOSUB 2730
800 REM 9000 dibujar
810 GOSUB 2590
820 REM decison de fin
830 IF p5=0 THEN CLS#2:CLS#4:GOTO 2
90 ELSE IF p5=2 THEN GOTO 870
840 CLS#2:CLS#4:PRINT#4,"pulsa F pa
ra acabar";PRINT#4,"pulsa S para s
eguir poniendo poleas";
850 e$=INKEY$
860 IF e$="f" OR e$="F" THEN GOTO 8
80 ELSE IF e$="s" OR e$="S" THEN CL
S#4:GOTO 290 ELSE GOTO 850
870 REM entrada de Masas,Angulos y
coef. de rozamiento
900 p2=0
910 FOR i=1 TO n1
920 CLS#2:CLS#4:PRINT#4,"Peso de la
Masa";i;" en Kg";:INPUT#4,masa(i)
930 SOUND 1,239,25,6:p1=0
940 PRINT#2,SPC(6);"Si quieres que
la masa este en";SPC(4);SPC(12);"e1
Suelo pulsa S";SPC(12);SPC(12);"Ve
rtical pulsa V";SPC(24);"un Plano p
ulsa P";SPC(12);"la decison limita
el movimiento de la M";
950 e$=INKEY$
960 IF e$="P" OR e$="p" THEN p2=p2+
1:p1=2 ELSE IF e$="V" OR e$="v" THE
N p1=1:p2=0 ELSE IF e$="S" OR e$="s
" THEN p1=4:p2=0 ELSE GOTO 950
970 IF p1=2 AND p2=2 THEN p1=3:p2=0
980 ka(i,3)=p1
990 IF p1=1 THEN ang(i)=90 ELSE IF
p1=2 OR p1=3 THEN CLS#4:INPUT#4,"An
gulo que forma el plano con el sue
lo",ang(i) ELSE IF p1=4 THEN ang(i)=
0
1000 IF ang(i)>90 OR ang(i)<0 THEN
GOTO 990

```

```

1010 IF masa(i)<0 THEN GOTO 920
1020 SOUND 1,190,20,6:NEXT i
1030 GOSUB 1100
1040 GOSUB 1630
1050 GOSUB 2190
1060 CLS#2:PRINT#2,"pulsa C para Co
menzar de nuevo          pulsa R pa
ra Resultados"
1070 e$=INKEY$
1080 IF e$="C" OR e$="c" THEN GOTO
2940 ELSE IF e$="R" OR e$="r" THEN
GOTO 1050 ELSE GOTO 1070
1090 END
1100 REM Esta subrutina crea y resu
lve un sistema de ecuaciones basad
o en el metodo de LAGRANGE
1110 REM Construccion de las fuerza
s en el sentido del movimiento y fu
erzas perpendiculares a el
1120 DEFREAL a,f:DIM fux(5),fuy(5),
fue(4),fur(4),mes(4)
1130 ERASE fux,fuy,fue,fur,mes
1140 FOR i=1 TO n1
1150 fux(i)=masa(i)*9.8*SIN(ang(i))
1160 fuy(i)=coef(i)*masa(i)*9.8*COS
(ang(i))
1170 NEXT i
1180 REM Construimos los vectores a
ce(i) a partir de pol(i..n4,i..n1) q
ue representaran la composicion de
las coordenadas absolutas en funcio
n de las generalizadas
1190 FOR i=n4 TO 1 STEP -1
1200 FOR j=1 TO n1
1210 IF pol(i,j)<0 THEN ace(ABS(pol
(i,j)),j)=-1
1220 IF pol(i,j)>0 THEN ace(pol(i,j
),j)=1
1230 NEXT j
1240 NEXT i
1250 REM Construimos la matriz A de
l sistema de ecuaciones A.x=b
1260 FOR i=1 TO n4
1270 FOR j=1 TO n4
1280 IF i<j THEN GOTO 1330

```

```

1290 FOR k=1 TO n1
1300 IF ace(k,i)=1 OR ace(k,i)=-1 T
HEN matriz(i,j)=matriz(i,j)+masa(k)
1310 NEXT k
1320 GOTO 1370
1330 FOR k=1 TO n1
1340 IF ace(k,i)=0 OR ace(k,j)=0 TH
EN GOTO 1360
1350 IF ace(k,i)=ace(k,j) THEN matr
iz(i,j)=matriz(i,j)+masa(k) ELSE IF
ace(k,i)=-1*ace(k,j) THEN matriz(i
,j)=matriz(i,j)-masa(k)
1360 NEXT k
1370 NEXT j
1380 REM Constrimos el vector fuerz
a en la direccion del movimiento
1390 FOR j=1 TO n1
1400 IF ace(j,i)=1 THEN fue(i)=fue(
i)+fux(j) ELSE IF ace(j,i)=-1 THEN
fue(i)=fue(i)-fux(j)
1410 NEXT j
1420 NEXT i
1430 IF n4=1 THEN acele(i)=fue(i)/m
atriz(i,i)
1435 GOSUB 3000
1440 REM Triangularizar la matriz
1450 FOR i=1 TO n4-1
1460 FOR j=i+1 TO n4
1470 FOR k=j TO n4
1480 matriz(j,i)=matriz(j,i)-(matr
iz(i,k)*matriz(j,i)/matriz(i,i))
1490 NEXT k
1500 fue(j)=fue(j)-(fue(i)*(matriz(
j,i)/matriz(i,i)))
1510 REM Compruebo a(i,i)<>0
1520 IF matriz(i+1,i+1)=0 THEN STOP
1530 NEXT j,i
1540 REM Calculamos las aceleracion
es
1550 FOR i=n4 TO 1 STEP -1
1560 IF i=n4 GOTO 1600
1570 FOR j=i+1 TO n4
1580 fue(i)=fue(i)-(matriz(i,j)*ace
le(j))
1590 NEXT j

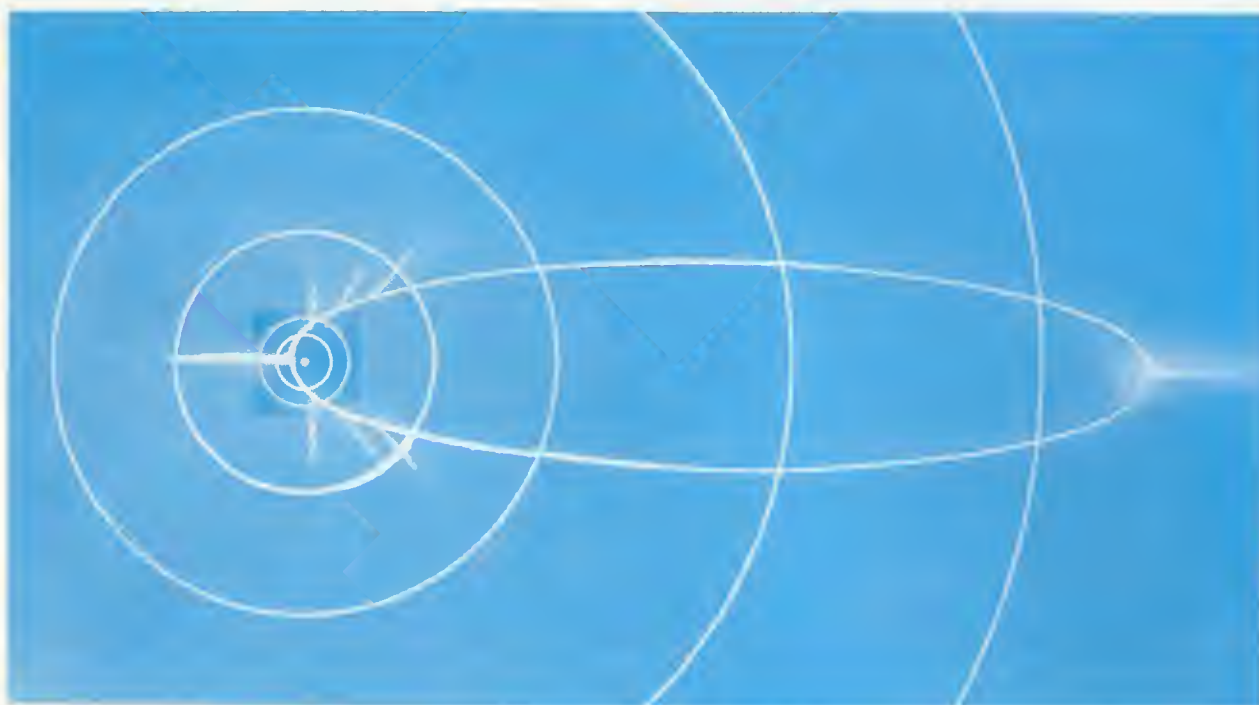
```



El programa podemos dividirlo en tres partes. En la primera, entran las restricciones impuestas al movimiento de las masas debidas a sus uniones mediante poleas. En la segunda, las restricciones al movimiento debidas al suelo en el que esté la masa, plano inclinado, suelo horizontal, caída libre, así como los datos necesarios de masa, ángulo... etc. Y en la tercera entra el cálculo propiamente dicho, y donde hemos utilizado una

## Mundo del CPC

simbolización para que derive el *lagrangiano* sin necesidad de derivar, basándonos en las formas únicamente cuadráticas que éste tiene.



```
1600 acele(i)=fue(i)/matriz(i,i)
1610 NEXT i
1615 GOSUB 3000
1620 RETURN
1630 REM pintar
1640 ca=FIX(640/n1):CLS:CLS#2:CLS#4
1650 FOR i=1 TO n1
1660 IF ang(i)<90 THEN GOTO 1750
1670 MOVE a(i)+8,12*16:TAG:PRINT CHR*(231);:TAGOFF
1680 MOVE a(i),9*16:DRAW a(i)+32,9*16:DRAW a(i)+32,9*16-25:DRAW a(i),9*16-25:DRAW a(i),9*16:MOVER 5,-5:FILL 1
1690 MOVE a(i)+16,12*16:DRAW 0,-3*16
1700 MOVE a(i)+16,9*16-25:DRAW 0,-20:PLOT a(i)+15,9*16-34:PLOT a(i)+16,9*16-34
1710 MOVE a(i),6*16:TAG:PRINT"M";i;:TAGOFF
1720 MOVE a(i)+30,9*16-34:TAG:PRINT"P";i;:TAGOFF
1730 Ka(i,1)=a(i)+8:Ka(i,2)=12*16
1740 GOTO 2050
1750 IF ang(i)>0 THEN GOTO 1850
1760 MOVE ca*(i-1),9*16-26:DRAW ca*i,9*16-26
1770 MOVE ca*(i-1)+(FIX((ca-48)/3)),9*16:DRAW 32,0:DRAW 0,-25:DRAW -32,0:DRAW 0,25:MOVER 5,-5:FILL 1
1780 MOVE ca*(i-1)+(2*FIX((ca-32)/3))+32,9*16-9:TAG:PRINT CHR*(231);:TAGOFF
1790 MOVE ca*(i-1)+(2*FIX((ca-32)/3))+32,9*16-9:TAG:PRINT CHR*(231);:TAGOFF
1800 MOVE ca*(i-1)+(FIX((ca-48)/3)),6*16:TAG:PRINT"M";i;:TAGOFF
1810 MOVE ca*(i-1)+(FIX((ca-48)/3))+16,9*16-25:DRAW 0,-20:PLOT ca*(i-1)+(FIX((ca-48)/3))+15,9*16-34:PLOT ca*(i-1)+(FIX((ca-48)/3))+17,9*16-34
```

```
1820 MOVE ca*(i-1)+(FIX((ca-48)/3))+30,9*16-34:TAG:PRINT"P";i;:TAGOFF
1830 Ka(i,1)=ca*(i-1)+(2*FIX((ca-32)/3))+40:Ka(i,2)=9*16-9
1840 GOTO 2050
1850 IF Ka(i,3)=2 THEN GOTO 1960
1860 MOVE ca*(i-1),12*16:DRAW ca*i,8*16
1870 MOVE FIX(ca*(i-1)+(ca-35)/2),12*16-FIX(((3*16/ca)*(ca-32)/2))-5:DRAW 5,(ca/(3*16))*5:DRAW 30,-((3*16/ca)*30):DRAW -5,-(ca/(3*16))*5:DRAW -30,((3*16/ca)*30):MOVER 5,5:FILL 1
1880 MOVE FIX(ca*(i-1)+(ca-35)/2),6*16:TAG:PRINT"M";i;:TAGOFF
1890 MOVE ca*(i-1)+8,13*16:TAG:PRINT CHR*(231);:TAGOFF
1900 MOVE ca*(i-1),12*16+8:DRAW FIX(ca*(i-1)+(ca-35)/2)+8,12*16-FIX(((3*16/ca)*(ca-32)/2))
1910 MOVE FIX(ca*(i-1)+(ca-35)/2)+15,12*16-FIX(((3*16/ca)*(ca-32)/2))-((3*16/ca)*30)-5:DRAW 0,-20:PLOT FIX(ca*(i-1)+(ca-35)/2)+14,12*16-FIX(((3*16/ca)*(ca-32)/2))-((3*16/ca)*30)-15
1920 PLOT FIX(ca*(i-1)+(ca-35)/2)+16,12*16-FIX(((3*16/ca)*(ca-32)/2))-((3*16/ca)*30)-15
1930 MOVE FIX(ca*(i-1)+(ca-35)/2)-17,11*16-FIX(((3*16/ca)*(ca-32)/2))-15:TAG:PRINT"P";i;:TAGOFF
1940 Ka(i,1)=ca*(i-1)+16:Ka(i,2)=13*16
1950 GOTO 2050
1960 MOVE ca*(i-1),8*16:DRAW ca*i,12*16
1970 MOVE FIX(ca*(i-1)+(ca-35)/2),8*16+FIX(((3*16/ca)*(ca-35)/2))+10:DRAW -5,(ca/(3*16))*5:DRAW 30,-((3*16/ca)*30):DRAW 5,-(ca/(3*16))*5:DRAW -30,-((3*16/ca)*30)
1980 MOVER 5,5:FILL 1
1990 MOVE FIX(ca*(i-1)+(ca-35)/2),6
```

```
*16:TAG:PRINT"M";i;:TAGOFF
2000 MOVE ca*i-16,13*16:TAG:PRINT CHR*(231);:TAGOFF
2010 MOVE ca*i,13*16-8:DRAW FIX(ca*(i-1)+(ca-35)/2)-3,8*16+FIX(((3*16/ca)*(ca-32)/2))+15
2020 MOVE FIX(ca*(i-1)+(ca-35)/2)+15,8*16+FIX(((3*16/ca)*(ca-32)/2))+((3*16/ca)*30+7:DRAW 0,-20:PLOT FIX(ca*(i-1)+(ca-35)/2)+14,8*16+FIX(((3*16/ca)*(ca-32)/2))+((3*16/ca)*30-2
2030 PLOT FIX(ca*(i-1)+(ca-35)/2)+16,8*16+FIX(((3*16/ca)*(ca-32)/2))+((3*16/ca)*30-2
2040 Ka(i,1)=ca*i-8:Ka(i,2)=13*16:MOVE FIX(ca*(i-1)+(ca-35)/2)+30,8*16+FIX(((3*16/ca)*(ca-32)/2))+((3*16/ca)*30-10:TAG:PRINT"P";i;:TAGOFF
2050 NEXT i
2060 FOR i=1 TO n4
2070 IF c(i,1)<n1 THEN v1=pol(c(i,1)-n1,6):u1=pol(c(i,1)-n1,7) ELSE v1=Ka(c(i,1),1):u1=Ka(c(i,1),2)
2080 IF c(i,2)<n1 THEN v2=pol(c(i,2)-n1,6):u2=pol(c(i,2)-n1,7) ELSE v2=Ka(c(i,2),1):u2=Ka(c(i,2),2)
2090 pol(i,6)=(v1+v2)/2
2100 IF u1>u2 THEN pol(i,7)=u1+32 ELSE pol(i,7)=u2+32
2110 NEXT i
2120 FOR i=1 TO n4
2130 MOVE pol(i,6),pol(i,7):TAG:PRINT CHR*(231);:TAGOFF
2140 IF c(i,1)<=n1 THEN MOVE pol(i,6)+8,pol(i,7):DRAW Ka(c(i,1),1),Ka(c(i,1),2) ELSE MOVE pol(i,6)+8,pol(i,7):DRAW pol(c(i,1)-n1,6),pol(c(i,1)-n1,7)
2150 IF c(i,2)<=n1 THEN MOVE pol(i,6)+8,pol(i,7):DRAW Ka(c(i,2),1),Ka(c(i,2),2) ELSE MOVE pol(i,6)+8,pol(i,7):DRAW pol(c(i,2)-n1,6),pol(c(i,2)-n1,7)
```

# Mundo del CPC

Una gran parte de la información entra y sale de forma gráfica para que los estudiantes de COU o primero que empiecen a ver este método, tengan una buena comprobación de los resultados por ellos obtenidos al trabajar con Lagrange.



```

2160 MOVE pol(i,6)+16,pol(i,7)-16:T
AG:PRINT"X";i;:TAGOFF
2170 NEXT i
2180 RETURN
2190 CLS#2:CLS#4
2200 FOR i=1 TO n1
2210 FOR j=1 TO n4
2220 IF ace(i,j)=1 THEN rez(i)=rez(
i)+acele(j) ELSE IF ace(i,j)=-1 THE
N rez(i)=rez(i)-acele(j)
2230 NEXT j
2240 NEXT i
2250 PRINT#4,"pulsa G aceleraciones
Generalizadas pulsa A acelerac
iones Absolutas pulsa C Con
tinuar"
2260 e$=INKEY$
2270 IF e$="a" OR e$="A" THEN GOTO
2290 ELSE IF e$="g" OR e$="G" THEN
GOTO 2320 ELSE IF e$="c" OR e$="C"
THEN GOTO 2350 ELSE GOTO 2260
2280 REM aceleraciones absolutas
2290 CLS#2
2300 FOR i=1 TO n1:PRINT#2,"ACELERA
CION de la masa";i;"=";rez(i):NEXT
i
2310 CLS#4:GOTO 2250
2320 CLS#2
2330 FOR i=1 TO n4:PRINT#2,"ACELERA
CION en cuerda";i;"=";acele(i):NEXT
i
2340 CLS#4:GOTO 2250
2350 RETURN
2360 c(n4+1,1)=n2:c(n4+1,2)=n3
2370 IF n2>n1 THEN m1=pol(n2-n1,6):
m2=pol(n2-n1,7) ELSE m1=a(n2):m2=9*
16
2380 IF n3>n1 THEN m3=pol(n3-n1,6):
m4=pol(n3-n1,7) ELSE m3=a(n3):m4=9*
16
2390 pol(n4+1,6)=(m1+m3)/2
2400 IF m4>m2 THEN pol(n4+1,7)=m4+3
2 ELSE pol(n4+1,7)=m2+32
2410 m5=0
2420 FOR i=1 TO n4
2430 IF pol(n4+1,6)=pol(i,6) AND po
l(n4+1,7)=pol(i,7) THEN m5=1
2440 NEXT i
2450 IF m5=1 THEN m5=0:pol(n4+1,7)=
(pol(n4+1,7)+32):GOTO 2420
2460 IF (n2 OR n3)>n1 THEN GOTO 248
0
2470 pol(n4+1,1)=n2:pol(n4+1,2)=-1*
n3
2480 IF n2<n1 THEN pol(n4+1,1)=n2:
m5=1:GOTO 2540
2490 m5=0
2500 FOR i=1 TO n1
2510 pol(n4+1,i)=ABS(pol(n2-n1,i))
2520 IF pol(n4+1,i)<>0 THEN m5=m5+1
2530 NEXT i
2540 IF n3<n1 THEN pol(n4+1,m5+1)=
-1*n3:RETURN
2550 FOR i=m5+1 TO n1
2560 pol(n4+1,i)=-1*ABS(pol(n3-n1,i
-m5))
2570 NEXT i
2580 RETURN
2590 m1=0:m2=0
2600 FOR i=1 TO n1
2610 IF pol(n4,i)=0 THEN m1=i
2620 IF b(i,2)=0 THEN m2=1
2630 NEXT i
2640 IF m1=0 THEN p5=2:GOTO 2670
2650 IF m2=0 THEN p5=1:GOTO 2670
2660 p5=0
2670 RETURN
2680 e$=INKEY$
2690 IF e$="C" OR e$="c" THEN p1=1
ELSE IF e$="D" OR e$="d" THEN p1=2
ELSE IF e$="O" OR e$="o" THEN p1=3
ELSE GOTO 2680
2700 RETURN
2710 MOVE m3,m4:TAG:PRINT f2$;TAGO
FF:MOVE m1,m2:TAG:PRINT f1$;TAGOFF
2720 RETURN
2730 REM pintar
2740 CLS:CLS#2:CLS#4
2750 FOR i=1 TO n1
2760 MOVE a(i),9*16:TAG:PRINT d$;T
AGOFF:MOVE a(i),8*16:TAG:PRINT d$;
TAGOFF:MOVE a(i),7*16:TAG:PRINT"m";
i;:TAGOFF
2770 NEXT i
2780 IF n4=0 THEN RETURN
2790 FOR i=1 TO n4
2800 o1=c(i,1):o2=c(i,2)
2810 MOVE pol(i,6),pol(i,7):TAG:PR
INT CHR$(231);:TAGOFF
2820 IF o1>n1 THEN m3=pol(o1-n1,6):
m4=pol(o1-n1,7) ELSE m3=a(o1)+16:m4
=9*16
2830 IF o2>n1 THEN m5=pol(o2-n1,6):
m6=pol(o2-n1,7) ELSE m5=a(o2)+16:m6
=9*16
2840 MOVE pol(i,6),pol(i,7):DRAW m5
,m6
2850 MOVE pol(i,6),pol(i,7):DRAW m3
,m4
2860 NEXT i
2870 RETURN
2880 IF na>n1+n4 THEN na=1
2890 IF na<n1 THEN GOTO 2920
2900 IF b(na,2)=1 THEN na=na+1:GOTO
2880
2910 RETURN
2920 IF pol(na-n1,8)/=1 THEN na=na+1
:GOTO 2880
2930 RETURN
2940 RUN
3000 MODE 2:CLS
3001 FOR i=1 TO n4
3010 FOR j=1 TO n4
3020 a=(40-n4*6)*(n4+2):d=(20-2*n4)
/(n4+1)
3030 LOCATE a*j+6*(j-1),i*d:PRINT m
atriz(i,j);"v";j
3040 NEXT j
3050 LOCATE a*(n4+1)+6*n4,i*d:PRINT
"=";fue(i)
3060 NEXT i
3070 LOCATE 1,23:PRINT"pulsa Cont
inuar"
3080 e$=INKEY$
3090 IF e$<"C" AND e$<"c" THEN GO
TO 3080
3100 MODE 1:WINDOW #4,1,40,22,25:PA
PER #4,3:CLS#4
3110 WINDOW #2,1,40,1,5:PAPER#2,3:C
LS#2
3120 RETURN

```



Actualidad, pokes, mapas, trucos,  
los mejores juegos y programas para  
**SPECTRUM, AMSTRAD, COMMODORE y MSX**

Todo el universo  
del Software  
mes a mes

---



**MICROMANÍA** ya está a la venta  
**¡Pídela en tu Kiosco!**



# *Reviews* **JUEGOS**

## **DONKEY KONG**

### **La bella y la bestia**

Erbe.

Cinta: 875 ptas.

Tel. 447 34 10

Disco:

Don Giulio tenía un enorme problema. *Donkey Kong*, el gorila del circo, había raptado a la bella Dolly, una de las trapezistas del circo, con la intención de vivir una aventura amorosa con ella. Esta claro que Dolly no quería saber nada de tan velludo pretendiente, por lo que Don Giulio se propuso rescatarla de lo alto de la carpa del circo, lugar donde la había colocado *Donkey Kong*. Éste, al ver cómo subía el propietario del circo a través del andamiaje, se empeñó en la nada pacífica tarea de arrojarle bidones de agua y petróleo. Esto suponía el tener que realizar la escalada mientras saltaba para evitar los bidones. Pero no acababan aquí las dificultades. Algunos de los bidones se incendiaban en más de una ocasión y ayudaban a caldear el ambiente. Más como este caballeroso señor no se arredraba fácilmente, rápidamente cogió un mazo que por allí colgado vio y, repartiendo mamporros a diestro y siniestro, logró romper los bidones y apagar los fuegos.

Bueno, ésta es la romántica historia de amor de *Donkey Kong* y la bella Dolly, pero como estamos seguros de que más de una persona estará dispuesta a ayudar a Don Giulio a rescatar a la trapezista, vamos a darles algunos consejos.

Para que los barriles no bajen por la escalera por la que subimos, será necesario sacar la mano justo por encima del nivel de la escalera.

No debemos salirnos de las plataformas puesto que perderemos una vida. Esto mismo ocurrirá si nos pasamos al saltar.

Cada objeto que saltemos nos dará 100 puntos,



si son dos 300 puntos, siendo 800 puntos si son tres los superados. Si en vez de saltar los golpeamos con el martillo, ganaremos 300, 500 u 800 puntos. En la cuarta pantalla ganaremos 100 puntos si quitamos los tornillos que encontraremos. El premio *gordo*, una vida extra, nos tocará cuando hayamos llegado a los 10.000 puntos.

Planteamiento simple, pero sumamente entretenido, el de *Donkey Kong*. Si añadimos la posibilidad de poder jugar con un amigo, los buenos ratos están asegurados.

Gráficamente no es sorprendente, pero está bien realizado y tiene un color agradable. Si a lo anteriormente expuesto, le añadimos un movimiento vivaracho, tendremos como producto final un juego realmente simpático.

*Donkey Kong* es de Ocean y es distribuido por Erbe.

**Digno de mención:** el tema. Simple, pero adictivo.





## UCHI MATA

# O cómo aprender judo por ordenador

Erbe. Tel. 447 34 10  
Cinta: 875 ptas. Disco:



Llegará un momento en que todo, o casi todo, se pueda aprender a través del ordenador. Sirva como ejemplo de este hecho, el juego de Martech, *Uchi Mata*. Con él podremos conocer y aprender a realizar todas las llaves de la milenaria arte marcial del judo. En un principio deberemos mostrar la máxima atención a la demo del juego, puesto que los dos contendientes realizarán todas las llaves antes mencionadas. A continuación, podremos jugar contra un amigo o contra el ordenador y... ¡¡Qué gane el mejor!!

Con un tema tan interesante y del que se podía haber sacado mucho más partido, la gente de Martech no ha logrado un resultado final excesivamente sorprendente. Gráficamente el juego flojo, falta definición, y la realización del color es mediocre, ya que en algunas ocasiones da la sensación de que se «va», de que se desplace de la figura. El movimiento no está mal pero es un tanto mecánico.

*Uchi Mata* es distribuido por Erbe.

**Digno de mención:** el tema.

**A mejorar:** el juego en general



## PALITRON

¡¡La ciudad  
está en peligro!!  
ponla a salvo  
con tu servo class 027

Erbe. Tel. 447 34 10  
Cinta: 875 ptas. Disco:

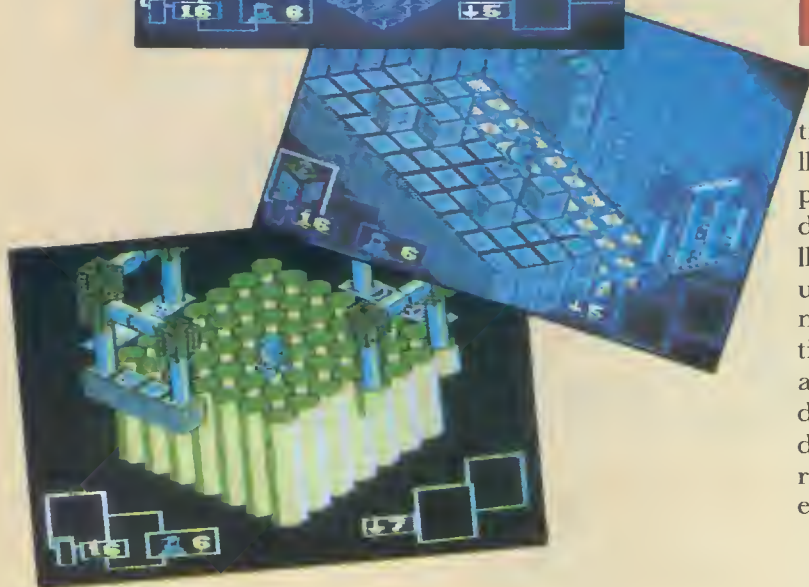
Como noble caballero del siglo XXI, deberás salvar la ciudad con tu rápido y poderoso robot *Servo Class 027* de los cristales explosivos que destruyen todo. Pero como esta misión representaría un peligro para tu integridad física, deberás programar y utilizar posteriormente los más pequeños robots *Infra Class Beast Mark 3*, para realizar misiones que serían prácticamente imposibles sin ellos. Los cristales podremos destruirlos si cogemos y tiramos sobre ellos, los objetos que encontremos a nuestro paso.

Una curiosidad destacable del juego es la posibilidad, ya mencionada anteriormente, de programar a los robots *Infra-Class*. Para hacer esto, necesitaremos una pila atómica de carga.



## Previews

# JUEGOS



El apartado gráfico, realizado con técnica 3D, cuenta con una buena definición y colores bien aplicados, sí como un movimiento rápido y preciso. En esta línea se encuentra también el sonido que ambienta el juego.

*Palitron* está distribuido por Erbe.

**Digno de mención:** El movimiento. Aúna rapidez y precisión.

### EXPLORER

## Busca y encuentra, la vida te va en ello

Procin S.A.

Tel. 276 22 08

Cinta: 880 ptas.

Disco: 2.695 ptas.

Tanto tiempo conduciendo naves de transporte, que el exceso de confianza te había llevado a una relajación de la vigilancia, y a no percatarte de cómo los estabilizadores inerciales de la nave se debilitan y se separan. Esto te había llevado fuera de tu ruta y acercado a la órbita de un planeta. Tal fue el impacto, que se partió en nueve trozos. Menos mal que pudiste salir a tiempo en uno de los módulos de supervivencia, y aterrizar en el planeta. Ahora te quedaba lo más duro; buscar en un planeta totalmente desconocido los cachos de tu nave y ser capaz de reconstruirla para volver a la civilización. Si bien esto ya era bastante difícil, lo que vendría a

Después, nos pondremos al lado de uno de ellos y pulsaremos la tecla CTRL, para posteriormente utilizar las teclas de cursor o joystick, con la finalidad de poder ver el menú y movernos por él.

Las cajas que hay en la parte inferior de la pantalla sirven o indican lo siguiente:

Las cuatro que se encuentran en la parte inferior de la pantalla están destinadas a almacenar los objetos que vayamos recogiendo.

La cajita pequeña en la parte inferior de la pantalla, sirve para mostrar la energía que tenemos.

Hay otra que indica el número de vidas con las que contamos.

En el centro derecha de la pantalla se nos muestra el número máximo de objetos que podemos soltar en esa pantalla.

Cuando hayamos soltado una bomba, un reloj nos indicará el tiempo que queda para que ésta estalle y así podremos saber el tiempo con que contamos para alejarnos.

Completo y bien realizado este juego de The Edge, que no dudamos que consiga hacer pasar buenos ratos a los jugadores y a los que no lo son tanto.







continuación pondría más emoción en esta tarea.

En *Explorer* debemos hacer gala de un buen sentido de la orientación y estrategia acercada.

Para movernos por el planeta deberemos utilizar joystick o los números del teclado alfanumérico. Nosotros recomendaríamos el teclado, ya que confiere mayor precisión. Cuando queramos coger un objeto, utilizar el láser, o el radio localizador, deberemos usar cualquier otra tecla.

Pero mientras buscamos, no debemos olvidar que hay peligros acechantes.

Siempre hemos opinado que sobre gustos no hay nada escrito, por lo que pensamos que

aunque para nosotros *explorer* es un tanto soso —demasiada búsqueda, poca acción—, seguro que habrá personas que este tipo de juegos, mezcla de estrategia y acertijo, les gustará.

Este juego es gráficamente aceptable, y aunque el color es un tanto pobre, no termina de desagradar. El movimiento, hecho enteramente por scroll, está bien. La cosa que más nos ha gustado de *Explorer* es el efecto zoom. De verdad que está bien realizado.

*Explorer* pertenece a Electri Deams y es distribuido por Proein S.A.

**Digno de mención:** el logrado efecto zoom.

**A mejorar:** el tema. Un poco soso.



## ACES OF ACES

### Demuestra que eres piloto

Edici. Tel. 90 14 10  
Cinta 875 ptas. Discos

Los simuladores de vuelo siempre han tenido un buen número de adeptos. De todas formas, la verdad sea dicha, este número de adeptos a acogido siempre mejor a los simuladores de aviones de combate, llenos de acción y posibilidades. *Ace of Aces* entra en este concepto lleno, y nos permite la posibilidad de conducir nuestro *mosquito* en varias incursiones diferentes, para hacer una demostración de nuestras cualidades de piloto y llegar a ser un as de ases. Pero antes de esto habrá que leerse el completo manual de controles de nuestro avión, para poder manejarlo apropiadamente. Por razones de espacio mencionaremos tan sólo los indicadores con que contamos y su función:



## Previews

# JUEGOS



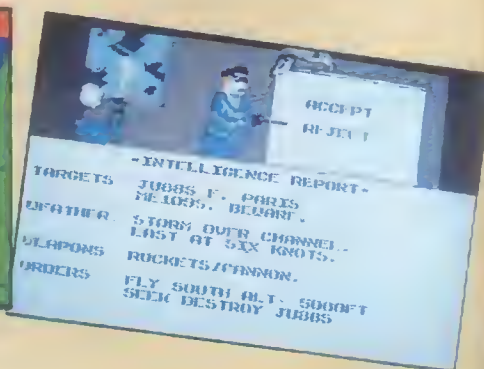
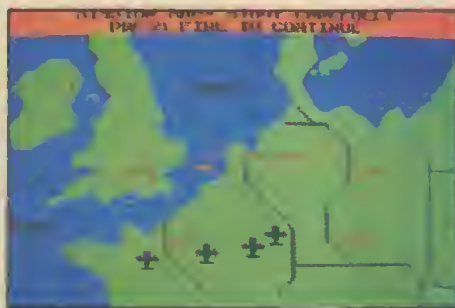
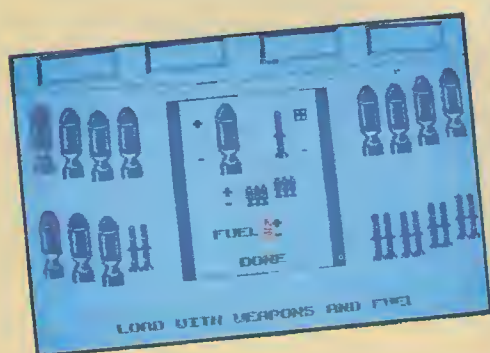
*Compás:* sirve para indicar nuestra dirección.

*Radar:* la pantalla giratoria a la derecha del todo, indica la distancia relativa del avión enemigo.

El indicador vertical indica la altura relativa del avión enemigo sobre la nuestra.

*Indicador de velocidad:* nos informa de la velocidad de vuelo.

*Indicador de altura:* evidentemente muestra a la



altura que nos hayamos. La podremos regular tirando o empujando el joystick.

*Horizonte artificial:* nos muestra el ángulo de nuestro avión en relación al horizonte.

*Palanca de motores:* sirve para regular las rpm (revoluciones por minuto del motor). Para hacer esto deberemos utilizar el joystick.

*Booster:* sirve para poder controlar la inclinación de las hélices.

*Extintor:* su uso es evidente. No obstante, cuidado con su utilización, los motores apagados por un extintor no pueden volver a encenderse.

*Trim:* controla el timón del avión y, por lo tanto, la dirección de vuelo.

*Tren de aterrizaje:* está siempre en posición de subida, pero será interesante el bajarlo en ocasiones para reducir la velocidad y poder esquivar a los aviones enemigos.

*Flaps:* sirven para variar la superficie de sustentación alar del avión. Modifican la altura y sirven para frenar. Cuidado con hacer bruscamente esto último.

Después de haber repasado los controles con los que contamos, pasaremos a continuación a enumerar las incursiones que deberemos realizar para llegar a ser un as de ases, éstas son:

*Bombardeo de trenes:* se trata de detener un tren de prisioneros que va camino de Berlín, y liberarlos. ¡¡Cuidado con bombardear los vagones de la Cruz Roja!!

*Bombardeo de submarinos:* los submarinos preparan un ataque a un convoy en el Atlántico norte. Deberás detenerlos antes de que salgan de sus bases.

*Lucha aérea:* cuidado con los cazas enemigos, aparecen al azar y son difíciles de derribar.

*Bombas V-1:* destrúyelas antes de que lleguen a Londres y lo reduzcan a cenizas.

Completo, por no decir completísimo, este simulador de vuelo de U.S. Gold. Pero como todo esto no serviría de nada sin no estuviera bien realizado, nosotros podemos afirmar que *Ace of Aces* lo está. Los gráficos son buenos, el color es el que se podría esperar en un simulador, al igual que el movimiento. Si para terminar añadimos que las acciones están bien desarrolladas, se puede decir sin duda que este simulador convencerá, como a nosotros, a más de una persona.

*Aces of Aces* es distribuido por Erbe.

**Digno de mención:** dentro de una línea bastante alta, el número de acciones bélicas tocadas.



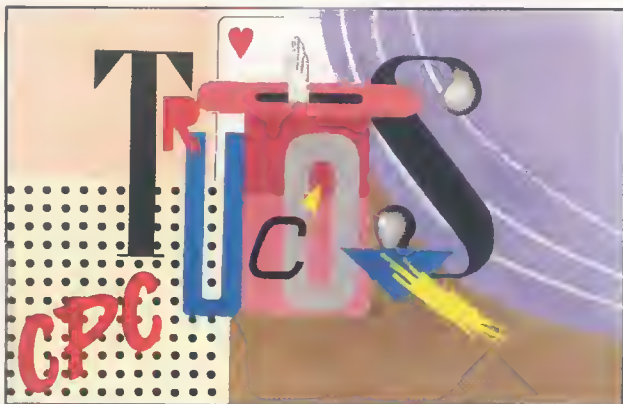
# COLECCION AMSTRAD!

**850 ptas.**

**Para solicitar  
las tapas,  
remítanos  
hoy mismo  
el cupón de pedido  
que encontrará  
en la solapa  
de la última página**



**No necesita encuadernación,  
gracias a un sencillo  
sistema de fijación  
que permite además  
extraer cada revista  
cuantas veces sea necesario.**



## Cambios en el programa

En muchas ocasiones, una vez terminado un programa, nos vemos en la tediosa obligación de variar una palabra por otra, y esto naturalmente a lo largo de toda la longitud de éste.

Esta tarea puede resultar de lo más tediosa cuando nos vemos en la obligación de realizarla sobre un buen número de palabras.

La utilización de esta utilidad es muy sencilla y sólo hay que seguir las instrucciones que se nos



soliciten. La única precaución que deberemos tomar es que el programa Basic deberá haber sido grabado previamente en ASCII, es decir, save «programa»,a, que es la forma que el locomotive emplea para este tipo de formato de archivo.

```
10 MODE 2
20 WINDOW #0,1,60,1,25:PEN #0,
0:PAPER #0,1
30 WINDOW #1,65,80,1,25
40 CLS #1:CLS #0
50 INPUT "PALABRA A CAMBIAR?."
";ELEC$
60 INPUT "CAMBIAR POR?...."
";CAMBIO$
70 INPUT "PROGRAMA A MODIFICAR
";PROG$
80 OPENIN prog$
90 OPENOUT prog$
100 WHILE NOT(EOF)
110 LINE INPUT #9,a$
120 PRINT #0,a$
130 IF INSTR(A$,ELEC$)>0 THEN
GOTO 170
140 PRINT #9,a$
150 WEND
160 END
170 PRINT #0,"----NUEVA LINEA-
----"
180 MID$(a$,INSTR(a$,elec$),LE
N(elec$))=cambio$
190 PRINT A$
200 GOTO 130
```

## Eliminar lo innecesario

Al terminar un programa, siempre quedan caracteres blancos al final de las líneas. Este programa realiza la labor de eliminarlos, con el consiguiente ahorro de memoria que esto supone.

A pesar de lo sencilla que resulta esta tarea, ésta se complica al no poder ver si realmente existen o no estos caracteres.

Al igual que el programa anterior, la labor se realiza tratando el programa Basic a modificar como un simple fichero de texto y para lo cual, deberemos haber grabado éste en modo ASCII.4.

```
10 INPUT "nombre del programa
basic.....";nom$
20 OPENIN nom$
30 OPENOUT nom$
40 WHILE NOT EOF
50 LINE INPUT #9,a$
70 char$=RIGHT$(a$,1)
80 IF char$=" " THEN a$=LEFT$
(a$,LEN(a$)-1):GOTO 70
100 PRINT #9,a$
110 WEND
120 CLOSEIN
130 CLOSEOUT
```





## Aperturas multicolores

Con este programa conseguiremos varias aperturas, que podremos fácilmente utilizar en nuestros propios programas.

Una vez ejecutado le ofrecerá la primera apertura gráfica en espera de la pulsación de una tecla, tras lo que comenzará la siguiente.

No nos cansaremos de repetir que el verdadero interés de estos programitas es incitarle de alguna forma a que realice sus propias modificaciones y que estudie los distintos efectos que éstas provocan.

Por ejemplo, le recomendamos que juegue con los sistemas de impresión, OR, XOR, AND; seguro que obtendrá buenos resultados.



```

10 CLS
20 DEFINT a-z
30 WHILE INKEY$=""
40 FOR n=1 TO 600 STEP 10
50 PLOT 0,n,1: DRAW 600,n
60 PLOT n,0: DRAW n,400
70 NEXT
80 FOR n=1 TO 600 STEP 10
90 PLOT 0,n,0: DRAW 600,n
100 PLOT n,0,0: DRAW n,400
110 NEXT
120 WEND
130 WHILE INKEY$=""
140 FOR n=1 TO 200 STEP 4
150 PLOT 0,400-n,1: DRAW 600,400-n
160 PLOT 0,n,1: DRAW 600,n
170 NEXT
180 FOR n=200 TO 1 STEP -4
190 PLOT 0,200+n,0: DRAW 600,200+n
200 PLOT 0,200-n,0: DRAW 600,200-n
210 NEXT
220 FOR n=1 TO 200 STEP 4
230 PLOT 0,200+n,1: DRAW 600,200+n
240 PLOT 0,200-n,1: DRAW 600,200-n
250 NEXT
260 WEND
    
```

## BOUTIQUE DEL ORDENADOR

AMSTRAD PCW ... 69.500  
AMSTRAD 464 .... 39.500

**Procedentes de exposiciones o cambios. Garantizados, impecables.**

**Consulte para unidades de disco, impresoras, etc...**

**ENVÍOS A TODA ESPAÑA**  
Tel.: (91) 416 13 02 (tardes)

## NECESITAMOS personas

que conozcan profundamente lenguaje ensamblador del 8086/8088 y dominen los ordenadores compatibles IBM PC desde el punto de vista del Lenguaje Máquina. Se valorarán en gran medida conocimientos acerca del DOS.

Si usted es uno de ellos póngase en contacto con nosotros rápidamente en la dirección que se indica a continuación, poniendo en lugar muy visible del sobre:

Referencia DOS AMSTRAD Semanal

Nuestra dirección es:

**AMSTRAD Semanal**

Carretera de Irún, km 12,400. Fuencarral. 28049 Madrid.

## ¿CUÁL ES SU TRUCO?

Desde esta columna queremos invitar a todos, sin excepción, a que participen en nuestra revista y se conviertan en parte aún más activa, si cabe, de la misma. Estamos convencidos que durante el uso diario de su ordenador han descubierto multitud de trucos y picares artimañas, maneras de hacer lo imposible, que han convertido horas de tedioso trabajo en momentos en los que la chispa de la inspiración le hacen a uno felicitarse por haber comprado un Amstrad. Si efectivamente nos los han descubierto, ¿por qué no nos los envían? Nosotros los miraremos todos, y los seleccionados recibirán como compensación por su esfuerzo cuatro cintas de cassette repletas de juegos, utilidades y programas publicados por **AMSTRAD Semanal**, de las que publicamos cada mes. Además, si el lector nos indica en la carta cuáles quiere, le enviaremos precisamente ésas, siempre que no estén agotadas.

Enviar las cartas a la siguiente dirección: **AMSTRAD Semanal**. Ctra. de Irún, Km. 12,400. 28049 Madrid.  
Indicando claramente en el sobre: **AMSTRAD Semanal**. Referencia Trucos.

# SEMINAR

A collage of various magazine covers and advertisements, including 'PCW', 'Ganar a Dólares', 'MSTR', and 'EXCLUSIVA', arranged in a circular pattern. The collage is set against a yellow background. The text 'PCW' is prominent in the upper left. Below it, 'GANAR A DÓLARES' is visible. In the center, 'MSTR' is written in large, bold letters. To the right, 'EXCLUSIVA' is partially visible. The collage also includes a small image of a person's face and some text in Spanish, such as 'DO EL SOFTWARE PARA EL' and 'Ganar a Dólares'.

# Solicita los números atrasados



# Una cinta gratis por la compra de números atrasados



Con la compra  
de siete números  
atrasados  
recibirás gratis  
una cinta de  
programas de  
**YOUR  
COMPUTER** (el  
mejor software  
inglés),  
totalmente  
gratis.

## Aprovecha la oferta y consigue tu colección



# Ahorrar memoria en gráficos (II)

Por Alberto Suñer

*Siguiendo con el tema de cómo incluir el mayor número posible de gráficos en nuestros programas, en esta ocasión presentamos una rutina con la que se consigue ahorrar la mitad de memoria en cada uno de ellos. Está especialmente indicada para aquellas imágenes que ocupen la mayor parte de la pantalla.*

**E**n algún capítulo anterior ya se ha abordado este problema, al cual se le han dado algunas posibles soluciones, hoy intentaremos dar otra salida a este gran inconveniente que en algunos casos nos obliga a rechazar gráficos verdaderamente bonitos debido a su gran longitud.

Con el método que comentaremos a continuación, estaremos en condiciones de ahorrar la mitad de la memoria que ocupe un gráfico, en cualquiera de los modos de pantalla en que nos permite trabajar nuestro **Amstrad**.

Las ventajas de la aplicación de esta rutina son verdaderamente asombrosas, ahora bien, esto no se consigue por arte de magia, y, por tanto, deberá existir algún inconveniente.

```
10 ORG      #9000
20
30
40 ;***MAGNIFICADDR DE GRAFICDS***
50
60
70 LD BC,TABLA
80 LD HL,ESPACE
90 JP #BCD1
100 TABLA:DEFW NAME
110 JP MAGN1
120 NAME:DEFM "MAGN"
130 DEFB     "I"+#80
140 DEFB     0
150 ESPACE:DEFS 4
160
170
180
```

```
190 MAGN1:LD  A,(IX+0)
200 LD (ANCHD),A
210 SLA      A
220 LD (ANCHD1),A
230 LD B,0
240 LD C,A
250 LD (LONG),BC
260 LD A,(IX+2)
270 SLA      A
280 SLA      A
290 SLA      A
300 LD (ALTO),A
310 ADD      A,A
320 LD (ALTD1),A
330 LD L,(IX+4)
340 LD H,(IX+5)
350 LD E,(IX+6)
360 LD D,(IX+7)
370 LD (BUFFER),HL
```

```
380 LD (GRAFIC),DE
390 LD L,(IX+8)
400 LD H,(IX+10)
410 LD (PDSIC),HL
420 LD A,(IX+12)
430 CP 1
440 JP Z,MAGN11
450 CP 0
460 JP Z,MAGN10
470 MAGN12:CALL MIRA2
480 CALL     IMPRE
490 RET
500 MIRA2:LD  DE,(BUFFER)
510 LD HL,(GRAFIC)
520 LD A,(ALTO)
530 LD B,A
540 KKK2:PUSH BC
550 LD (PDSIN),DE
560 LD A,(ANCHD)
570 LD B,A
580 KKK1:XOR  A
590 BIT      7,(HL)
600 CALL     NZ,PONG7
610 BIT      6,(HL)
```





# Gráficos por ORDENADOR

El método utilizado para el ahorro de memoria, consiste en aumentar el tamaño del gráfico por dos, es decir, igual que si lo viéramos a través de una lupa, con lo que, obviamente, se pierde algo de resolución gráfica.

Como ya hemos dicho anteriormente, esta pérdida de resolución puede resultar mínima, si se aplica a figuras relativamente grandes, ya que en ellas queda mucho más disimulada la pérdida de resolución.

Por tanto, aconsejaríamos que este método se utilizara únicamente para gráficos que estén

**El mayor problema con los gráficos es la memoria que ocupan.**

destinados a ocupar gran parte de la pantalla, aunque se puede aplicar a figuras de dimensiones más reducidas con resultados verdaderamente muy buenos.

La utilización de esta rutina, es especialmente recomendada para aplicarla en aquellos gráficos que deban ocupar gran parte de la pantalla, ya que se conseguirá rellenar gran parte de la misma con una escasa ocupación de memoria, con lo que se podrán obtener pantallas más vistosas al estar ocupadas la mayor parte de ellas por gráficos.

Vamos a explicar a continuación cómo funciona cada una de las rutinas para los diferentes modos de pantalla.



Para el modo de alta resolución, el funcionamiento es bien sencillo, únicamente se trata de duplicar cada uno de los bits que componen un byte, con lo que se conseguirá dos bytes de gráfico aumentado por cada uno de gráfico original.

Esto se conseguirá chequeando los bits, y en caso de que estén a uno, duplicarlos con unos, y en caso de estar a ceros duplicarlos con ceros.

Dicha operación se realizará para cada línea horizontal. A continuación se duplicarán línea por línea, con lo que se obtendrá la magnificación en sentido vertical.

La rutina que se encarga de realizar este trabajo, para el modo 1 de pantalla, es similar a la anterior, pero en este caso deberemos tener en cuenta que los bits llevan, en este caso, información del color con que se va a imprimir el gráfico, por lo que se deberá tratar de una forma especial.

Así pues, deberemos tener en cuenta cada una de las parejas de bits que se encuentran en un byte.

```

620 CALL      NZ,PON45
630 BIT       5,(HL)
640 CALL      NZ,PON23
650 BIT       4,(HL)
660 CALL      NZ,PON01
670 LD (DE),A
680 XOR       DE
690 INC       3,(HL)
700 BIT       NZ,PON67
710 CALL      2,(HL)
720 BIT       NZ,PON45
730 CALL      1,(HL)
740 BIT       NZ,PON23
750 CALL      0,(HL)
760 BIT       NZ,PON01
770 CALL
780 LD (DE),A
790 INC       DE
800 INC       HL
810 DJNZ      KKK1
820 PUSH      HL
830 LD HL,(POSIN)
840 LD BC,(LONG)
850 LDIR
860 POP       HL
870 POP       BC
880 DJNZ      KKK2
890 LD (POSIN),DE
900 RET
910 MAGN11:CALL MIRA1
920 CALL      IMPRE
930 RET
940 MIRA1:LD  DE,(BUFFER)
950 LD HL,(GRAFIC)
960 LD A,(ALTO)
970 LD B,A
980 LLL2:PUSH BC
990 LD (POSIN),DE
1000 LD A,(ANCHD)

```

```

1010 LD B,A
1020 LLL1:XOR  A
1030 BIT      7,(HL)
1040 CALL     NZ,PON67
1050 BIT      6,(HL)
1060 CALL     NZ,PON45
1070 BIT      3,(HL)
1080 CALL     NZ,PON23
1090 BIT      2,(HL)
1100 CALL     NZ,PON01
1110 LD (DE),A
1120 XOR      A
1130 INC      DE
1140 BIT      5,(HL)
1150 CALL     NZ,PON67
1160 BIT      4,(HL)
1170 CALL     NZ,PON45
1180 BIT      1,(HL)
1190 CALL     NZ,PON23
1200 BIT      0,(HL)

```

En este caso, igual que en el anterior, tomaremos en primer lugar cuatro bits originales con lo que se logrará un byte ampliado y luego los otros cuatro.

Ahora bien, los bits que deberemos tomar en primer lugar serán aquellos que lleven la información de los dos primeros pixels, que serán los bits 7, 6 con sus correspondientes parejas, el 3 y 2.

El siguiente byte ampliado se formará con las restantes parejas de bits.

Para el modo de menor resolución, es decir el que permite mayor número de colores simultáneamente, la operación de magnificación será algo distinta a las anteriores, ya que en este caso el byte original únicamente es capaz de imprimir dos pixels.

Así pues, el byte aumentado que se obtenga imprimirá únicamente un pixel. Por lo tanto, en este caso se observará la información correspondiente al primer pixel, la cual se indica en alguno de los bits 7, 5, 3, 1 y se ampliará, consiguiéndose así el primer byte ampliado.



```

1210 CALL      NZ,PON01
1220 LD (DE),A
1230 INC       DE
1240 INC       HL
1250 DJNZ      LLL1
1260 PUSH      HL
1270 LD HL,(POSIN)
1280 LD BC,(LONG)
1290 LDIR
1300 PDP       HL
1310 PDP       BC
1320 DJNZ      LLL2
1330 LD (PDSIN),DE
1340 RET
1350 MAGNIO:CALL      MIRAO
1360 CALL      IMPRE
1370 RET
1380 MIRAO:LD      DE,(BUFFER)
1390 LD HL,(GRAFIC)
1400 LD A,(ALTD)
1410 LD B,A

```

```

1420 MMM2:PUSH BC
1430 LD (POSIN),DE
1440 LD A,(ANCHO)
1450 LD B,A
1460 MMM1:XOR    A
1470 BIT       7,(HL)
1480 CALL      NZ,PON67
1490 BIT       5,(HL)
1500 CALL      NZ,PON45
1510 BIT       3,(HL)
1520 CALL      NZ,PON23
1530 BIT       1,(HL)
1540 CALL      NZ,PON01
1550 LD (DE),A
1560 XOR        A
1570 INC        DE
1580 BIT       6,(HL)
1590 CALL      NZ,PON67
1600 BIT       4,(HL)
1610 CALL      NZ,PON45
1620 BIT       2,(HL)

```

**La rutina reduce la ocupación de una figura a la mitad.**

El restante byte magnificado se conseguirá chequeando la información de los restantes bits.

Debemos tener en cuenta que dichas operaciones deberán efectuarse para todos y cada uno de los bytes que componen el gráfico original.

Estas rutinas almacenarán el gráfico ampliado a partir de la posición de memoria que nosotros indiquemos, de esta forma conoceremos la localización de nuestro nuevo gráfico.

El programa que hemos preparado incorpora una rutina de impresión para mostrarnos la forma del gráfico una vez magnificado, aunque dicha rutina se podrá quitar en cualquier momento, con sólo eliminar las llamadas que se hacen a la misma desde distintas partes del programa.

Debemos tener en cuenta que el gráfico obtenido finalmente, tendrá una longitud doble a la del gráfico original, cosa que deberemos tener presente en caso de que se quiera salvar en disco o cinta dicha figura.

Para realizar esta operación únicamente deberemos efectuar un *save* con la dirección que hayamos indicado en la correspondiente llamada al programa, y una longitud doble a la del gráfico original.

Para un manejo más fácil de la rutina de magnificación de gráficos, hemos preparado un nuevo comando *RSX*, con el cual podremos realizar más fácilmente las operaciones de ampliación de gráficos.

Esta nueva instrucción será del siguiente tipo:

```

1630 CALL      NZ,PON23
1640 BIT       0,(HL)
1650 CALL      NZ,PON01
1660 LD (DE),A
1670 INC        DE
1680 INC        HL
1690 DJNZ      MMM1
1700 PUSH      HL
1710 LD HL,(POSIN)
1720 LD BC,(LONG)
1730 LDIR
1740 POP        HL
1750 PDP        BC
1760 DJNZ      MMM2
1770 LD (POSIN),DE
1780 RET
1790 PDSIN:DEFS 2
1800 PON01:SET  0,A
1810 SET        1,A
1820 RET
1830 PON23:SET  2,A

```



# Gráficos por ORDENADOR

IMAGNI,MOD,POSX,POSY,GRAF,DIRB,  
DIMX,DIMY

donde cada uno de los parámetros corresponde a:

*Mod* . . . . . Modo de pantalla del gráfico.  
*Posx* . . . . . Coordenada vertical de pantalla,  
imprimirá el gráfico obtenido.  
*Posy* . . . . . Coordenada horizontal.  
*Graf* . . . . . Dirección del gráfico original.  
*Dirb* . . . . . Dirección del gráfico ampliado.  
*Dimx* . . . . . Altura del gráfico original.  
*Dimy* . . . . . Anchura del gráfico original.

De esta forma una vez ejecutada la rutina de magnificación, el gráfico ampliado se encontrará en la dirección indicada por el parámetro *dirb*, con lo cual lo podremos salvar de la forma indicada anteriormente.

Debemos tener cuidado con el valor que se da a dicho parámetro, ya que un valor erróneo podría provocar un *cuelgue* del ordenador. Esto ocurrirá si dicha dirección se encuentra cerca de la dirección de ejecución del programa, es decir de la posición de memoria &8000.

Así pues, para que no ocurra nada anormal, la dirección del gráfico ampliado, deberá ser lo más baja posible para que de esta forma no pueda interrumpir en ningún momento nuestro programa.

**El programa se especializa  
en gráficos de gran  
tamaño.**



Debemos fijarnos también en el valor que se le da a la variable *mod*, ya que dando un valor erróneo, el resultado obtenido será inservible, puesto que para cada modo de pantalla se utilizan diferentes rutinas.



1840 SET 3,A  
1850 RET  
1860 PON45:SET 4,A  
1870 SET 5,A  
1880 RET  
1890 PON67:SET 6,A  
1900 SET 7,A  
1910 RET

1920 ;RUTINA-IMPRESION  
1930 ;H-POSICION-VERTICAL-INICIO-1  
1940 ;L-POSICION-HORIZONTAL-INICIO-1  
1950 ;DE-DIRECCION-GRAFICO  
1960 ;  
1970 IMPRE:LD HL,(POSIC)  
1980 LD DE,(8BUFFER)  
1990 PUSH DE  
2000 EX DE,HL  
2010 LD HL,&C000-80  
2020 LD B,D  
2030 LD D,0  
2040 DEC E

2050 ADD HL,DE  
2060 LD DE,80  
2070 S\_BUC:ADD HL,DE  
2080 DJNZ S\_BUC  
2090 POP IX  
2100 LD A,(ALTO1)  
2110 LD B,A  
2120 JR COLDC  
2130 P\_BUC:LD A,H  
2140 AND 56  
2150 CP 56  
2160 JR Z,P\_PAS  
2170 LD A,H  
2180 ADD A,8  
2190 LD H,A  
2200 JR COLDC  
2210 P\_PAS:LD DE,0080  
2220 LD A,H  
2230 XOR 56  
2240 LD H,A  
2250 ADD HL,DE

2260 COLDC:PUSH BC  
2270 PUSH HL  
2280 LD A,(ANCH01)  
2290 LD B,A  
2300 P\_BUC1:LD A,(IX+0)  
2310 LD (HL),A  
2320 INC IX  
2330 INC HL  
2340 DJNZ P\_BUC1  
2350 POP HL  
2360 POP BC  
2370 DJNZ P\_BUC  
2380 RET  
2390 ALTO:DEFS 1  
2400 ANCHO:DEFS 1  
2410 ALTO1:DEFS 1  
2420 ANCHO1:DEFS 1  
2430 POSIC:DEFS 2  
2440 LONG:DEFS 2  
2450 GRAFIC:DEFS 2  
2460 BUFFER:DEFS 2



## Gráficos por ORDENADOR



**Gracias al Código Máquina  
no se aprecia disminución  
alguna de velocidad.**

Los valores a introducir en los parámetros correspondientes a la dimensión del gráfico, deberán darse en caracteres, tanto en alto como en ancho.

Para aquellos que deseen separar las rutinas de ampliación de gráficos de la rutina de impresión de pantalla, diremos que las correspondientes, a cada uno de los modos son las siguientes:

*Magni0* .. Ampliación para modo 0  
*Magni1* .. Ampliación para modo 1  
*Magni2* .. Ampliación para modo 2

Para la utilización de esta rutina, deberemos copiar el listado ensamblador que aparece al final del artículo, y una vez ensamblado salvarlo en cinta o disco.

Aquellos que no posean ensamblador pueden

utilizar el cargador Basic que aparece a continuación, y una vez ejecutado, en caso de no existir errores, almacenarlo de la forma siguiente:

SAVE "MAGNI",B,&9000,&1DE

Cuando deseemos ejecutar el programa, deberemos cargar el código objeto en la dirección de memoria &9000, realizando una llamada a dicha dirección para inicializar los nuevos comandos *RSX*.

A partir de ese momento estaremos en disposición de ampliar los gráficos que deseemos en cualquiera de los modos de pantalla.

### PROGRAMA CARGADOR

|                                       |                                      |                                      |
|---------------------------------------|--------------------------------------|--------------------------------------|
| 1 REM AHORRAR MEMORIA EN GRAFICOS(11) | 240 DATA 175,203,126,196,145,145,203 | 500 DATA 237,83,128,145,58,223,145   |
| 2 REM PROGRAMA CARGADOR               | 250 DATA 118,196,140,145,203,110,196 | 510 DATA 71,175,203,126,196,145,145  |
| 10 FOR N=&9000 TO &91DE               | 260 DATA 135,145,203,102,196,130,145 | 520 DATA 203,110,196,140,145,203,94  |
| 20 READ A:SUMA=SUMA+A                 | 270 DATA 18,175,19,203,94,196,145    | 530 DATA 196,135,145,203,78,196,130  |
| 30 POKE N,A                           | 280 DATA 145,203,86,196,140,145,203  | 540 DATA 145,18,175,19,203,118,196   |
| 40 NEXT                               | 290 DATA 78,196,135,145,203,70,196   | 550 DATA 145,145,203,102,196,140,145 |
| 50 IF SUMA<>65991 THEN PRINT          | 300 DATA 130,145,18,19,35,16,207     | 560 DATA 203,86,196,135,145,203,70   |
| "ERROR EN DATAS"                      | 310 DATA 229,42,128,145,237,75,228   | 570 DATA 196,130,145,18,19,35,16     |
| 60 DATA 1,9,144,33,20,144,195         | 320 DATA 145,237,176,225,193,16,184  | 580 DATA 207,229,42,128,145,237,75   |
| 70 DATA 209,188,14,144,195,24,144     | 330 DATA 237,83,128,145,201,205,201  | 590 DATA 228,145,237,176,225,193,16  |
| 80 DATA 77,65,71,78,201,0,0           | 340 DATA 144,205,150,145,201,237,91  | 600 DATA 184,237,83,128,145,201,0    |
| 90 DATA 0,0,0,221,126,0,50            | 350 DATA 232,145,42,230,145,58,222   | 610 DATA 0,203,199,203,207,201,203   |
| 100 DATA 223,145,203,39,50,225,145    | 360 DATA 145,71,197,237,83,128,145   | 620 DATA 215,203,223,201,203,231,203 |
| 110 DATA 6,0,79,237,67,228,145        | 370 DATA 58,223,145,71,175,203,126   | 630 DATA 239,201,203,247,203,255,201 |
| 120 DATA 221,126,2,203,39,203,39      | 380 DATA 196,145,145,203,118,196,140 | 640 DATA 42,226,145,237,91,232,145   |
| 130 DATA 203,39,50,222,145,135,50     | 390 DATA 145,203,94,196,135,145,203  | 650 DATA 213,235,33,176,191,66,22    |
| 140 DATA 224,145,221,110,4,221,102    | 400 DATA 86,196,130,145,18,175,19    | 660 DATA 0,29,25,17,80,0,25          |
| 150 DATA 5,221,94,6,221,86,7          | 410 DATA 203,110,196,145,145,203,102 | 670 DATA 16,253,221,225,58,224,145   |
| 160 DATA 34,232,145,237,83,230,145    | 420 DATA 196,140,145,203,78,196,135  | 680 DATA 71,24,21,124,230,56,254     |
| 170 DATA 221,110,8,221,102,10,34      | 430 DATA 145,203,70,196,130,145,18   | 690 DATA 56,40,6,124,198,8,103       |
| 180 DATA 226,145,221,126,12,254,1     | 440 DATA 19,35,16,207,229,42,128     | 700 DATA 24,8,17,80,0,124,238        |
| 190 DATA 202,194,144,254,0,202,33     | 450 DATA 145,237,75,228,145,237,176  | 710 DATA 56,103,25,197,229,58,225    |
| 200 DATA 145,205,106,144,205,150,145  | 460 DATA 225,193,16,184,237,83,128   | 720 DATA 145,71,221,126,0,119,221    |
| 210 DATA 201,237,91,232,145,42,230    | 470 DATA 145,201,205,40,145,205,150  | 730 DATA 35,35,16,247,225,193,16     |
| 220 DATA 145,58,222,145,71,197,237    | 480 DATA 145,201,237,91,232,145,42   | 740 DATA 216,201,0,0,0,0,0           |
| 230 DATA 83,128,145,58,223,145,71     | 490 DATA 230,145,58,222,145,71,197   |                                      |



SOMOS MAYORISTAS

# MICRO-1

EL IVA  
LO PAGA MICRO-1

C/Duque de Sesto, 50. 28009 Madrid  
Tel. (91) 275 96 16 - 274 75 02

Metro O'Donnell o Goya (aparcamiento gratuito en Felipe II)

SOFTWARE:  
POR CADA DOS PROGRAMAS,  
GRATIS A ELEGIR  
- CASCOS STEREO  
- RELOJ DIGITAL + BOLIGRAFO  
- LACADO  
- CALCULADORA EXTRAPLANA

|                    | PTAS. |                  | PTAS. |
|--------------------|-------|------------------|-------|
| FIST II            | 875   | XEVIOUS          | 875   |
| DEEP STRIKE        | 875   | 10th FRAME       | 1200  |
| SUPER SOCCER       | 875   | LEADERBOARD      | 1200  |
| TERRA CREST        | 875   | EXPRESS RAIDER   | 875   |
| DOUBLE TAKE        | 875   | ACE OF ACES      | 1200  |
| SHORT CIRCUIT      | 875   | IMPOSSABALL      | 875   |
| GAUNTLET           | 875   | SIGMA 7          | 875   |
| ARMY MOVES         | 875   | BAZZOKA BILL     | 875   |
| BREAKTHRU          | 875   | DRAGON'S LAIR II | 875   |
| 4 SUPER 4          | 1750  | SHADOW SKIMMER   | 875   |
| ¡¡NOVEDADES KONAMI |       | 1850 PTS!!       |       |

## IMPRESORAS 20% DESCUENTO SOBRE P.V.P.

|                       | PTAS. |
|-----------------------|-------|
| DISKETTE 3"           | 735   |
| DISKETTE 5 1/4" DC/DD | 295   |
| LÁPIZ ÓPTICO SPECTR   | 2890  |
| LÁPIZ ÓPTICO AMSTRAD  | 3290  |
| CINTA C-15 ESPEC.     | 69    |
| MICRODRIVE            | 495   |
| ARCHIVADOR DISCOS     | 2600  |

CASSETTE ESPECIAL ORDENADOR 3.495 PTS. Y 3.995 PTS

COMPATIBLE PC-IBM 640 K  
2 BOCAS 360 K  
MONITOR FÓSFORO VERDE  
149.900 PTS. (incl. IVA)

SOLICITA GRATIS  
NUESTRO CATÁLOGO A  
TODO COLOR, DE  
NUESTROS PRODUCTOS

|                              | PTAS.  |
|------------------------------|--------|
| SANYO MSX 64                 | 28.900 |
| COMMODORE 128                | 54.900 |
| COMMODORE 128 + TECL MUSICAL | 57.900 |

SERVICIO TÉCNICO REPARACIÓN TARIFA FIJA: 3.600 PTS  
(incl. provincias sin gastos envío)

SPECTRUM PLUS + CASCOS MÚSICA STEREO  
19.800 PTS (incl. IVA).

### CABLES E INTERFACES 20% DTO. SOBRE P.V.P.

CADENA MUSICAL 27.900 PTS.  
VIDEO VHS AKAI 79.900 PTS.  
RADIOCASSETTE STEREO 6.895 PTS.

|                    |   |
|--------------------|---|
| AMSTRAD 464 VERDE  | ENTRADA 7.000 PTS. 12 MESES A 4.900 PTS.  |
| AMSTRAD 464 COLOR  | ENTRADA 9.800 PTS. 12 MESES A 7.500 PTS.  |
| AMSTRAD 6128 VERDE | ENTRADA 8.900 PTS. 12 MESES A 7.182 PTS.  |
| AMSTRAD 6128 COLOR | ENTRADA 14.900 PTS. 12 MESES A 9.900 PTS. |

12 MESES CON EL 0% DE INTERÉS. ¡¡MICRO-1 TE LO FINANCIA GRATIS!!

RATÓN PARA AMASTRAD Y COMMODORE CON SOFTWARE 6.900 PTS.

PEDIDOS CONTRA REEMBOLSO SIN NINGÚN GASTO DE ENVÍO  
LLAMA POR TELÉFONO. ADELANTAS TRES DÍAS TU PEDIDO  
TELE. (91) 274 75 02 / (91) 275 96 16 (DURANTE LAS 24 HORAS)

TIENDAS Y DISTRIBUIDORES, PIDAN LISTA DE PRECIOS AL MAYOR.  
C/ GALATEA, 25. TELE. (91) 274 75 03

|                     | PTAS. |
|---------------------|-------|
| QUICK SHOT II       | 1.395 |
| QUICK SHOT II TURBO | 2.795 |
| QUICK SHOT IX       | 1.995 |
| KONIX (microswitch) | 2.595 |
| INTERFACE SPECTRUM  | 1.395 |

## **HARMA**

de Proa

P.V.P.: 67.200

Ofrece un interesante programa de cálculo de estructuras de hormigón armado, que dará acceso a los profesionales de la arquitectura e ingeniería a la resolución, mediante ordenador, de los antes complicados y largos cálculos de pórticos.

La resolución mediante cálculo matricial permite una mayor fiabilidad y seguridad en los cálculos, así como un gran ahorro de tiempo.

Por otra parte, la mayor seguridad en los cálculos, comporta una notable reducción de costos de obra para el constructor.

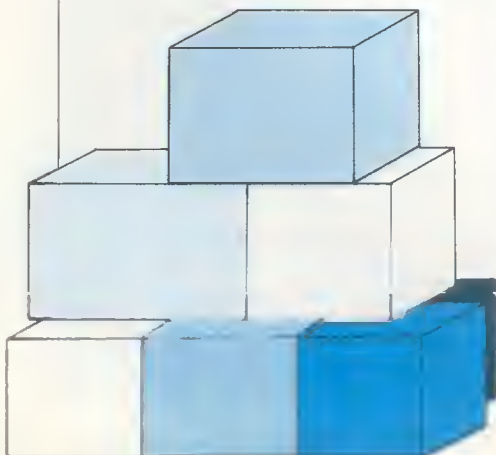
El programa *Harma* calcula cualquier pórtico plano de nudos rígidos con vigas y pilares de sección constante, por lo que resulta ideal para el cálculo de pórticos de hormigón armado.

Utiliza el método matricial de las deformaciones, determinando tanto las debidas a flexión como las de esfuerzos axiales.

### **1. Capacidad:**

La capacidad depende de la máxima diferencia de numeraciones entre nudos extremos de la barra.

Como dato orientativo podemos decir que en una



# Catálogo de **SOFTWARE PCW**

retícula sin ningún tipo de modificación (que la máxima diferencia entre extremos de barras sea igual a los vanos más uno) admite alrededor de cien nudos.

Así, por ejemplo, un pórtico de cuatro vanos, admitiría veinte pisos, uno de tres vanos, treinta pisos, etcétera.

## **2. Entrada de datos:**

### **a. Geométricas:**

La entrada de datos se realiza mediante un editor gráfico a partir de la generación de una retícula de vanos y plantas que sirve de base para el diseño final de la estructura.

No tenemos ningún tipo de restricción respecto a la geometría, por lo que podremos fijar hasta barras inclinadas, muy útiles para incluir, en la retícula de vigas y pilares, escaleras y cubiertas inclinadas.

Tampoco hay restricciones en vínculos externos, que pueden ser empotramientos, articulaciones, apoyos horizontales o verticales, así como voladizos.

### **b. Cargas:**

*Harma* admite hasta diez hipótesis simples con un número máximo de diez combinaciones entre ellas, con un número de cargas prácticamente ilimitado para cada hipótesis.

Por ejemplo, en una estructura de 178 nudos, admite un máximo de 480

cargas en barras y 480 cargas en nudos.

## **3. Salida de resultados:**

### **a. Hipótesis simples.**

El programa nos da los desplazamientos de los nudos en centímetros y sus giros en grados así como los esfuerzos intermedios en todas las barras en toneladas y metros.

Proporciona también las reacciones en los nudos soportes (generalmente las zapatas) por el método de equilibrio de fuerzas en nudos.

### **b. Combinaciones de hipótesis:**

Opcionalmente pueden darnos los esfuerzos y armado, en centímetros cuadrados, en cinco secciones por barra.

Estudia la combinación más favorable de hipótesis y da el armado, en centímetros cuadrados, en todas las barras.

Dibuja el armado longitudinal y transversal de las vigas en la combinación más desfavorable de hipótesis. Además, podemos elegir y cambiar la escala a la que salen los resultados.

La representación de la viga se hace en sección tanto longitudinal como transversal, con la representación de anclajes y montajes de armaduras según la EH-82.

## **CLUBS DE VÍDEO**

de Megsoft

P.V.P.: Sin confirmar.

La amplia difusión de que gozan los Vídeo-Clubs se ha visto facilitada por los programas dirigidos a estos comercios, cuyo trabajo de control y seguimiento es demasiado intenso.

Por ello, como ya en otras ocasiones hemos comentado en este catálogo de PCW, es inestimable la ayuda prestada



por estos programas a los Vídeo-Clubs, en los que el usuario tan sólo requiere elegir aquel programa cuyas prestaciones mejor se adecúen a sus requerimientos.

De aquí nuestro interés de ofrecer la más detallada información sobre las posibilidades que nos ofrece cada producto.

La presente aplicación ha sido desarrollada para llevar todo el control de socios, películas, movimiento y realización automática de recibos, de acuerdo con los siguientes criterios:

1. *Fichero de socios:*
  - Datos personales.
  - Fecha de inicio como socio.
  - Tipo de sistema de vídeo.
  - Datos bancarios para el caso de que el recibo esté domiciliado.
  - Altas, bajas, consultas y modificaciones.
2. *Control de los movimientos* de cada cliente, presentándose en pantalla todos ellos por el nombre del cliente, código de la película, fecha de la entrega y de devoluciones de la misma.
3. *Fichero de películas:*
  - Altas, bajas, consultas y modificaciones.
  - Listados.
4. *Realización de recibos*, que incluyen los siguientes conceptos:
  - Importe.
  - Vencimiento.
  - Datos pago cliente.
  - Penalizaciones.
5. *Listados y consultas por pantalla e impresora:*

El ordenador calcula las penalizaciones para los casos de socios de propiedad o de alquiler. En ambos casos, aparece el cargo en el recibo y un comentario sobre dicho cargo.

5. *Listados y consultas por pantalla e impresora:*

- Listados de socios.
- Penalizaciones.
- Diario de los movimientos de películas de propiedad y de alquiler.
- Consulta sobre tenencia de películas.
- Listado de películas.

6. *Realización de etiquetas:*

Esta opción nos permite agilizar diversos trámites, como por ejemplo el envío de circulares, empleando los datos que hemos almacenado en el programa.

## **CONTROL DENTISTA**

de Novus Software

P.V.P.: 29.192 ptas.

Este programa está dedicado a los profesionales de la estomatología y odontología, es decir, a los dentistas.

El objetivo es reducir el tiempo que el médico tiene que dedicar a la gestión de ficheros, dándole una mayor fluidez y organización, para que pueda dedicar un tiempo mayor al ejercicio estricto de su profesión.

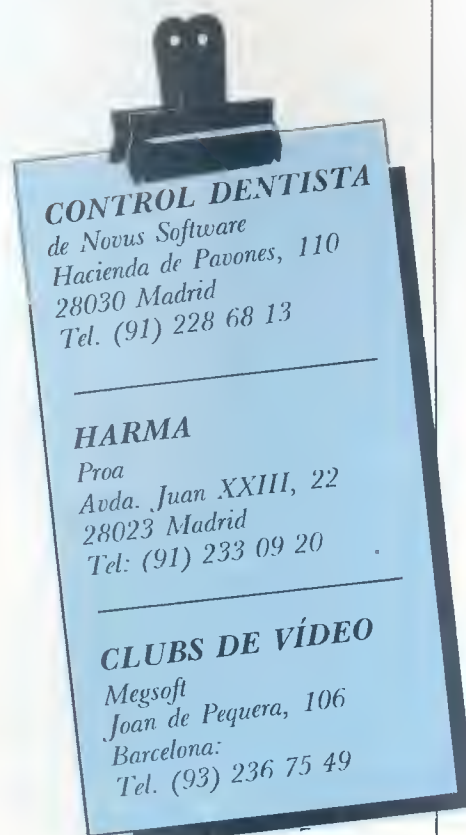
Lo mejor es que no se necesita tener ningún conocimiento de informática para manejar este programa. Su utilización es muy sencilla. Es más, el mismo programa va guiando e impidiendo que se cometan errores.

Funciona mediante un sencillo sistema de fichas que el médico puede clasificar de acuerdo con los criterios que considere oportunos y siempre con la posibilidad de una rápida y sencilla modificación. De la ficha principal de cada paciente cuelgan las de nueva visita, las de prótesis, etc. De esta manera tan sencilla, se agilizan el tiempo y el espacio dedicados a los ficheros y su localización.

Todos los datos se actualizan automáticamente, desde los económicos hasta la edad del paciente.

Una de las características más importantes del programa, desde el punto de vista de un profesional, es que las fichas de prótesis se adaptan a las normas odontológicas y, además, se han realizado, gráficamente, de tal forma, que con una rápida ojeada permiten conocer la composición de la prótesis, en cuanto a número y disposición de las piezas.

Un aspecto importante son los listados que emite, en los que aparece la ficha completa del paciente, con su saldo económico incluido.



# ¡SENSACIONAL CONCURSO MUSICAL DE AMSTRAD SEMANAL!

¡Puede ganar un equipo de alta fidelidad con Compact Disc!



*Las posibilidades sonoras de los ordenadores Amstrad están fuera de toda duda. Se puede hacer música con ellos y buena. Por esta razón, AMSTRAD Semanal organiza este concurso, ofreciendo como premio a los ganadores un equipo musical de alta fidelidad de la marca Investrónica, modelo CD-300 hf, con Compact Disc.*



## BASES

1.— El concurso se divide en tres categorías diferentes. Cada concursante puede optar por una de ellas, por las dos o por las tres si lo desea.

**Primera categoría:** consiste en la creación de un tema libre. Se puede enviar una melodía, canción o cualquier tipo de música.

**Segunda categoría:** el concursante deberá enviar los siguientes sonidos:

- sonido de llamada telefónica
- ruido de explosión
- sonido de un disparo
- ruido de arranque de un coche o moto
- sonido que produce el despegue de una nave espacial

**Tercera categoría:** esta categoría se reserva en exclusiva a los PCW por la clara inferioridad en que se encuentran respecto al resto de ordenadores de la gama **Amstrad**. Aquí se premiará la mejor melodía según lo expresado en la primera categoría o los mejores cinco sonidos que se especifican en la segunda. Por lo tanto, los concursantes pueden enviar melodías, sonidos o ambas cosas a la vez.

2.— Las melodías y/o los sonidos, en las tres categorías deberán enviarse a la redacción de **AMSTRAD Semanal** en forma de programa, bien en Basic, en Lenguaje Máquina o en programa cargador a base de líneas Data. No se admitirán ficheros binarios, o una mezcla de ambos lenguajes. El soporte será una cinta de cassette o un disco, en el formato requerido por el modelo de ordenador **Amstrad** para el que se haya escrito.

3.— El disco o cassette deberá enviarse a la siguiente dirección:

CONCURSO MUSICAL  
AMSTRAD SEMANAL  
APARTADO 226  
ALCOBENDAS  
MADRID

4.— Este concurso está abierto a todos los usuarios de ordenadores de la gama **Amstrad**, sin excepción, cualquiera que sea el modelo.

5.— El plazo de envío de las melodías y sonidos para el concurso comienza a partir del día 10 de febrero y finalizará el día 10 de mayo de 1987. Las cartas con matasellos posteriores a esta fecha no entrarán en concurso.

6.— Cada concursante puede enviar, dentro del plazo, una sola carta conteniendo su

creación. Si decide optar por ambos temas, la melodía libre y los cinco sonidos, deberá enviarlo todo junto en el mismo sobre, convenientemente separados para facilitar su identificación.

7.— El jurado estará formado por don Francisco Pastor del Pueyo (director de Erbe software), don Carlos Toro Montoro (compositor) y don José Luis Arriaza Ovrán (locutor del programa «Sábado chip» de la cadena **COPE**).

8.— Cada uno de los ganadores de las tres categorías recibirán un equipo de alta fidelidad «Investrónica CD-300 ht». No podrá recaer más de un premio en la misma persona. Para ello el jurado comenzará calificando por la primera categoría, luego la segunda y después la tercera. El concursante que resulte ganador en la primera categoría será eliminado, una vez que el jurado pase a calificar la segunda, si es que también ha participado en ella, y así en los siguientes casos.

Los premios serán enviados mediante agencia de transporte puerta a puerta (transporte por carretera para la península y marítimo para las islas).

9.— Se regalarán tres equipos de alta fidelidad repartidos entre las tres categorías:

- a la mejor melodía
- a los mejores cinco sonidos
- a la mejor melodía o mejores cinco sonidos en PCW.

10.— El jurado se reunirá y valorará las creaciones de los concursantes durante la primera quincena de mayo de 1987. De los resultados se informará cumplidamente en la revista. La decisión del jurado será protocolizada por un notario y cualquier lector podrá solicitar el acta notarial siempre que lo haga dentro de los tres meses posteriores a su fecha de emisión.

11.— Los agraciados recibirán comunicación personal por correo certificado de **AMSTRAD Semanal**.

12.— Todos los programas enviados por los concursantes quedan en propiedad de la revista **AMSTRAD Semanal**, reservándose el derecho de publicación si lo considerara oportuno.

13.— La participación en este concurso implica la aceptación total de todas las bases. Cualquier supuesto que se produzca, no especificado en estas bases, será resuelto por **HOBBY PRESS, S.A.**



# Sopa de letras

Por: Antonio Aguilera

*La informática es una asignatura que cada vez se va a revelar de mayor importancia para nuestro futuro y el de nuestra familia. Por eso, los programas que combinan lo entretenido con el aprendizaje merecen la pena. El programa Sopa de letras está encaminado en esta dirección. A la vez que constituye un reto a la retentiva y agilidad visual, las palabras que hay que descubrir forman una serie de términos constantemente utilizados en informática. Familiarizarse con ellos es un primer paso para llegar a conocer de esta pronto, imprescindible disciplina.*

**E**ste juego es la clásica sopa de letras en la que hay que buscar la palabra que aparece en la parte superior derecha de la pantalla.

Al empezar el juego, se debe introducir el número de palabras a buscar en cada partida. Una vez que se ha escondido ese número de palabras en la sopa (una palabra en cada sopa), aparecen los porcentajes de aciertos y errores.

Una vez encontrada la palabra escondida, se

debe introducir la columna y la fila en la que se encuentra la primera letra de la palabra (si la fila es menor que 10, hay que poner un cero delante). A continuación, mediante las teclas de cursor, la dirección en que está escondida la palabra, teniendo en cuenta que no estará en diagonal. Para ello se dispone de un tiempo de 20 segundos. Todas las palabras que pueden ser escondidas están relacionadas con la informática.

## VARIABLES

|         |   |
|---------|---|
| na      | Número de aciertos.                             |
| nf      | Número de fallos.                               |
| njp     | Número de palabras por partida.                 |
| p\$( )  | Matriz con las palabras.                        |
| mp\$( ) | Matriz de la sopa de letras.                    |
| p\$     | Palabra a buscar.                               |
| bft     | Bandera de tiempo acabado.                      |
| x,y,cf  | Coordenadas y dirección de la palabra a buscar. |
| c,f,dir | Coordenadas y dirección dadas por el jugador.   |
| l       | Longitud de la palabra.                         |

## ESTRUCTURA

|           |                                   |
|-----------|-----------------------------------|
| 10-210    | Rutina principal.                 |
| 250-700   | Inicialización.                   |
| 740-1030  | Inicialización.                   |
| 1070-1180 | Esconde palabra.                  |
| 1220-1360 | Pantalla.                         |
| 1400-1650 | Respuesta del jugador.            |
| 1690-1730 | Comprueba respuesta.              |
| 1770-1850 | Porcentajes.                      |
| 1890-1910 | Cronómetro.                       |
| 1950      | Fin de tiempo.                    |
| 1990-2350 | Subrutinas para esconder palabra. |



Serie

ORO

```

10 REM *****
20 REM *      SOPA DE LETRAS      *
30 REM *****
40 REM *      Antonio Fernando    *
50 REM *      Aguilera Romera     *
60 REM *      GRANADA.            *
70 REM *****
80 GOSUB 250: ' titulo
90 GOSUB 740: ' inicializacion
100 na=0:nf=0:nop=0
110 WHILE nop<njp
120 nop=nop+1
130 GOSUB 1070: ' esconde palabra
140 GOSUB 1220: ' pantalla
150 GOSUB 1400: ' respuesta
160 GOSUB 1690: ' comprueba respuest
a
170 WEND
180 GOSUB 1770: ' porcentajes
190 np$="":WHILE np$(">"S" AND np$(">"N":np$=UPPER$(INKEY$):WEND
200 MODE 1:INK 0,13:INK 1,0:INK 2,2
:INK 3,24:BORDER 13
210 IF np$="S" THEN 100 ELSE CLEAR:
END
220 REM *****
230 REM *      TITULO      *
240 REM *****
250 MODE 1
260 INK 0,13:INK 1,0:INK 2,13:INK 3
,24:BORDER 13
270 PAPER 0:PEN 1:CLS:CLG 0
280 WINDOW 2,39,2,24
290 PAPER 0:CLS
300 PEN 2
310 LOCATE 1,23:PRINT "SOPA DE LETR
AS"
320 LOCATE 1,1
330 FOR x=16 TO 16*15 STEP 2:FOR y=

```

```

16 TO 32 STEP 2
340 IF TEST(x,y)=2 THEN PLOT x+195,
(y+150)*2,3:DRAW 0,2:PLOT 4,4,1:D
RAW 0,-2
350 NEXT:NEXT
360 LOCATE 1,23:PRINT STRING$(14,32
)
370 INK 2,3
380 WINDOW 1,40,1,25
390 x=1:y=1:ancho=38:alto=24
400 PEN 2:LOCATE x,y:PRINT CHR$(213
);STRING$(ancho,143);CHR$(212)
410 PEN 3:PAPER 2:LOCATE x,y:PRINT
CHR$(223):LOCATE x+ancho+1,y:PRIN
T CHR$(222)
420 PAPER 3:PEN 2
430 FOR n=1 TO alto-1
440 LOCATE x,y+n:PRINT CHR$(207):LO
CATE x+ancho+1,y+n:PRINT CHR$(207)
450 NEXT
460 PEN 3:LOCATE x,y+alto:PRINT CHR
$(214);STRING$(ancho,143);CHR$(215)
;
470 PEN 2:PAPER 3:LOCATE x,y+alto:P
RINT CHR$(220):LOCATE x+ancho+1,y+a
lto:PRINT CHR$(221);
480 WINDOW 2,39,2,24
490 PLOT x*16-16,400-y*16+14,1:DRA
W (ancho+2)*16-2,0
500 DRAW 0,-16*(alto+1)+2:DRAW - (
ancho+2)*16+2,0
510 DRAW 0,16*(alto+1)-2:DRAW 14,
-14
520 DRAW ancho*16+2,0:DRAW 0,-16*
(alto-1)-2
530 DRAW -ancho*16-2,0:DRAW 0,16*
(alto-1)+2
540 PLOT ancho*16+2,0:DRAW 14,14
550 PLOT -14,-(16*alto+2):DRAW 14
,-14

```

```

560 PLOT -(ancho+1)*16-2,14:DRAW
-14,-14
570 ORIGIN 112,112,112,520,112,286:
CLG 2
580 FOR i=0 TO 416 STEP 20:PLOT 416
-i,0:DRAW 0,i/2.36:PLOT i,0:DRAW 41
6,i/2.36:PLOT 416-i,176:DRAW 0,176-
i/2.36:PLOT 416-i,176:DRAW 416,i/2.
36:NEXT
590 PAPER 0:PEN 3:LOCATE 1,20:PRINT
STRING$(38,154):LOCATE 1,22:PRINT
STRING$(38,154);
600 PEN 1:LOCATE 2,21:PRINT USING "
&";CHR$(174);"Numero de palabras po
r partida? ";
610 njp$=""
620 a$="":WHILE a$(">"CHR$(13) AND LE
N(njp$)<4
630 a$=INKEY$:IF a$="" THEN 630
640 IF (ASC(a$)>47 AND ASC(a$)<58)
THEN PRINT a$;njp$=njp$a$
650 IF ASC(a$)=127 AND LEN(njp$) TH
EN njp$=RIGHT$(njp$,LEN(njp$)-1):PR
INT CHR$(8);CHR$(16);
660 WEND
670 njp$=VAL(njp$)
680 IF njp<1 OR njp>12 THEN LOCATE
1,21:PRINT STRING$(38,32);PEN 1:LO
CATE 1,20:PRINT STRING$(38,154):LOC
ATE 1,22:PRINT STRING$(38,154):PEN
3:LOCATE 14,21:PRINT "ENTRE 1 Y 12"
:FOR r=1 TO 1500:NEXT:LOCATE 1,21:P
RINT STRING$(38,32):GOTO 590
690 LOCATE 1,21:PRINT STRING$(38,32
):PEN 1:LOCATE 1,20:PRINT STRING$(3
8,154):LOCATE 1,22:PRINT STRING$(38
,154):PEN 3:LOCATE 10,21:PRINT "NUM
ERO DE PALABRAS:";MID$(STR$(njp),2)
700 RETURN
710 REM *****
720 REM *      INICIALIZACION      *
730 REM *****
740 RANDOMIZE TIME
750 DIM p$(114)
760 FOR i=1 TO 114
770 READ p$(i)
780 NEXT
790 x(1)=8:y(1)=13:x(2)=8:y(2)=17:x
(3)=6:y(3)=15:x(4)=10:y(4)=15

```



```

800 DIM mp$(15,12)
810 SYMBOL AFTER 245
820 SYMBOL 246,255,254,252,248,240,
224,192,128
830 SYMBOL 247,255,255,0,0,0,0,0,0
840 SYMBOL 248,128,192,224,240,248,
252,254,255
850 SYMBOL 249,0,0,0,0,0,0,255,255
860 SYMBOL 250,255,255,0,0,0,0,0,0
870 SYMBOL 251,255,127,63,31,15,7,3
,1
880 SYMBOL 252,0,0,0,0,0,0,255,255
890 SYMBOL 253,1,3,7,15,31,63,127,2
55
900 SYMBOL 254,192,192,192,192,192,
192,192,192
910 SYMBOL 255,3,3,3,3,3,3,3,3
920 cu$=CHR$(10)+STRING$(2,8)
930 f1$(1)=CHR$(253)+CHR$(248)+cu$+
CHR$(255)+CHR$(254)
940 f1$(2)=CHR$(255)+CHR$(254)+cu$+
CHR$(251)+CHR$(246)
950 f1$(3)=CHR$(253)+CHR$(249)+cu$+
CHR$(251)+CHR$(250)
960 f1$(4)=CHR$(249)+CHR$(248)+cu$+
CHR$(250)+CHR$(246)
970 RETURN
980 DATA DIMENSION,MUNITOR,TECLADO,
AMSTRAD,PROGRAMA,GRAFICOS,SONIDO,PE
RIFERICO,IMPRESORA,ORDENADOR,INTERF
ACE,DISCO,SALIDA,CURSOS,JOYSTICK,CO
NECTOR,CASSETTE,CINTA,CADENA,PROCES
ADOR
990 DATA COLOR,NUMERO,LETRA,NUMERIC
O,TINTA,BURDE,FUSFORO,ACCESO,ARCHIV
O,DATOS,LECTURA,SISTEMA,CHIP,SILICI
O,DISKETTE,SIMULADOR,CENTRONICS,PAN
TALLA,MATRIZ,ESCRITURA,GRABACION,IN
FORMATICA,MEMORIA,BASIC,LOGO
1000 DATA PASCAL,FORTH,ENSAMBLADOR,
LENGUAJE,CALCULO,GESTION,APLICACION
,SOFTWARE,HARDWARE,CARACTER,BINARIO
,DECIMAL,HEXADECIMAL,PROGRAMADOR,AN
ALISTA,BUCLE,CONDICION,ERROR,MENSAJ
E,ORGANIGRAMA,ENVOLVENTE
1010 DATA ORDINOGRAMA,Puerta,AUDIO,
VIDEO,CANAL,VENTANA,OPERADOR,PERSO
NA,LINEA,COMANDO,FUNCION,VIDEOJUEGO
,AVENTURA,RESOLUCION
1020 DATA SUPORTE,EDITOR,TARJETA,AL
IMENTADOR,AMPLIACION,BUS,EXPANSION,
COMPATIBLE,SUBROUTINA,APRENDIZAJE,AC
UPLADOR,UNIDAD,LOGICA,GENERADOR,DIG
ITAL,ANALOGICO,CODIGO,RUTINA,ACUMUL
ADOR
1030 DATA ALEATORIO,SECUENCIAL,OPER
ATIVO,TRACCION,FRICCION,INDEXADOS,I
NTERCAMBIO,USUARIO,CONEXION,TRATAMI
ENTO,TEXTOS,RESULTADO,ALMACEN,CONFR
OL,BLOQUE

```

```

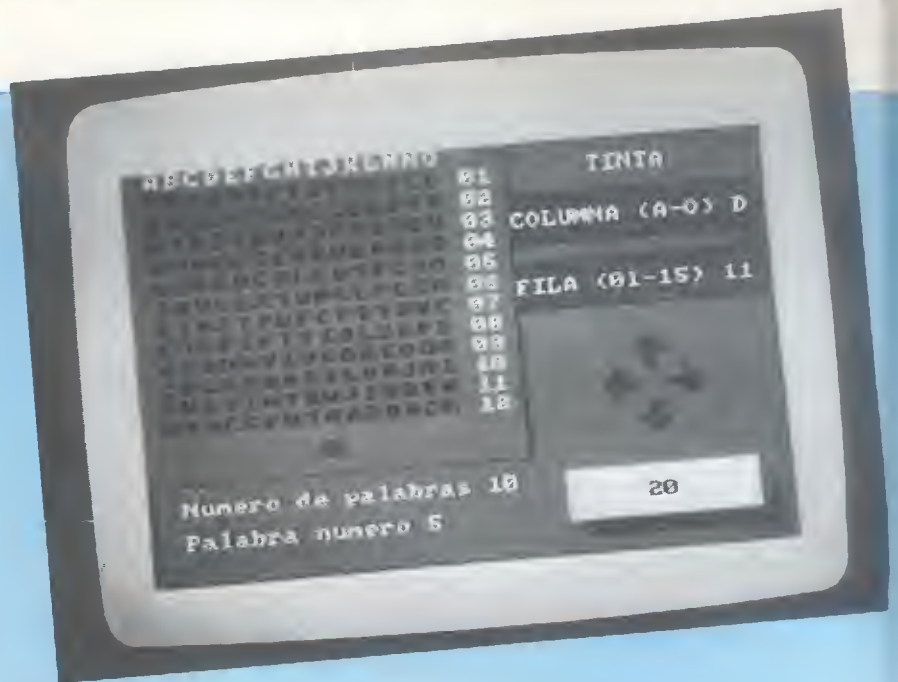
1040 REM *****
1050 REM * ESCONDE PALABRA *
1060 REM *****
1070 p$=p$(IN1(RND*114+1))
1080 FOR i=1 TO 15:FOR j=1 TO 12
1090 mp$(i,j)="
1100 NEXT:NEXT
1110 f=0:c=0:dir=0
1120 l=LEN(p$)
1130 cf=INT(RND*4+1)
1140 ON cf GOSUB 2290,2190,2090,199
0
1150 FOR i=1 TO 15:FOR j=1 TO 12
1160 IF mp$(i,j)=" THEN mp$(i,j)=C
HR$(IN1(RND*26+65))
1170 NEXT:NEXT
1180 RETURN
1190 REM *****
***
1200 REM * PREPARA PANTALLA DE JUEG
O *
1210 REM *****
***
1220 CLG 0:MODE 1:INK 2,2
1230 WINDOW #0,1,40,1,25:PAPER #0,0
:CLS:PEN #0,1:WINDOW #1,25,40,1,25:
PAPER #1,2:CLS #1:PEN #1,3
1240 WINDOW #6,1,24,20,25:PAPER #6,
2:PEN #6,3:CLS #6
1250 LOCATE #1,FIX(9-LEN(p$)/2),2:P
RINT #1,p$
1260 INK 1,13
1270 FOR j=1 TO 12:FOR i=1 TO 15:TA
G:PLUT i*20,399-j*20,1:PRINT mp$(i,
j):NEXT:TAGOFF:PRINT:NEXT
1280 var=0:FOR i=1 TO 15:TAG:PLUT i
*20,399,3:PRINT CHR$(65+var):var=v
ar+1:NEXT:TAGOFF
1290 num=1:FOR j=1 TO 12:TAG:PLUT 3
20,399-j*20,3

```

```

1300 IF num<10 THEN num$=" 0"+RIGHT
$(STR$(num),1):PRINT num$: ELSE PRI
NT num:
1310 num=num+1:NEXT:TAGOFF
1320 LOCATE 12,18
1330 INK 1,0
1340 WINDOW #3,28,39,22,24:PAPER #3
,1:CLS #3:WINDOW #2,27,38,23,21:PAP
ER #2,3:CLS #2:PEN #2,1:WINDOW #4,2
5,40,4,6:PAPER #4,1:CLS #4:PEN #4,3
:WINDOW #5,25,40,8,10:PAPER #5,1:CL
S #5:PEN #5,3
1350 PEN #1,1:FOR i=1 TO 4:LOCATE #
1,x(i),y(i):PRINT #1,f1$(i):NEXT
1360 RETURN
1370 REM *****
*
1380 REM * RESPUESTA DEL JUGADOR
*
1390 REM *****
*
1400 LOCATE #6,3,2:PRINT #6,"Numero
de palabras";injp
1410 LOCATE #6,3,4:PRINT #6,"Palabr
a numero";inop
1420 LOCATE #4,1,2:PRINT #4,"COLUMN
A (A-U) ";
1430 LOCATE #5,1,2:PRINT #5,"FILA (
01-15) ";
1440 PLOT 0,0,1:DRAW 638,0:DRAW 0
,398:DRAW -256,0:DRAW 0,-304:DRAW
R -382,0:DRAW 0,-96:PLUT 382,96:
DRAW 256,0
1450 bft=0
1460 tb=TIME
1470 GOSUB 1890
1480 EVERY 50,3 GOSUB 1890
1490 AFTER 1000,2 GOSUB 1950
1500 f=0:WHILE (f<1 OR f>15) AND bf
t=0
1510 f$=UPPER$(INKEY$)

```





```

1520 f=ASC(f$+CHR$(0))-64
1530 WEND
1540 PRINT #4,f$
1550 c=0:WHILE (c<1 OR c>12) AND bf
t=0
1560 c1$="":WHILE (c1$<"0" OR c1$>
"1") AND bft=0:c1$=UPPER$(INKEY$):WE
ND
1570 c2$="":WHILE (c2$<"0" OR c2$>
"9") AND bft=0:c2$=UPPER$(INKEY$):WE
ND
1580 c=VAL(c1$)*10+VAL(c2$)
1590 WEND
1600 IF c<10 THEN c$="0"+RIGHT$(STR
$(c),1) ELSE c$=RIGHT$(STR$(c),2)
1610 IF c>0 THEN PRINT #5,c$
1620 c3$="":WHILE (c3$<CHR$(240) OR
c3$>CHR$(243)) AND bft=0:c3$=UPPER
$(INKEY$):WEND
1630 IF c3$<>" " AND bft=0 THEN dir=
ASC(c3$)-239:PEN #1,3:LOCATE #1,x(d
ir),y(dir):PRINT #1,f$(dir)
1640 cr1=REMAIN(3):cr2=REMAIN(2)
1650 RETURN
1660 REM *****
**
1670 REM * COMPRUEBA LA RESPUESTA
*
1680 REM *****
**
1690 IF f=x AND c=y AND dir=cf THEN
FOR i=400 TO 200 STEP -1:SOUND 1,1
,1:NEXT:PRINT:PRINT TAB(8)CHR$(175)
"CORRECTO!":na=na+1:GOTO 1720
1700 IF bft=1 THEN PRINT:PRINT TAB(
5)"ESTABA EN ";CHR$(64+x);"-";RIGHT
$(STR$(y),LEN(STR$(y))-1);" ";CHR$(
239+cf) ELSE FOR i=200 TO 400:SOUND
1,1,1:NEXT:PRINT:PRINT TAB(3)"ERRO
R:ESTA EN ";CHR$(64+x);"-";RIGHT$(S
TR$(y),LEN(STR$(y))-1);" ";CHR$(239+
cf)
1710 nf=nf+1
1720 FOR wx=x TO nx STEP SGN(slx):F
OR wy=y TO ny STEP SGN(sly):TAG:PLD
T wx*20,399-wy*20,3:PRINT mp$(wx,wy
):FOR rr=1 TO 200:NEXT:NEXT:NEXT:TA
GUFF
1730 RETURN
1740 REM *****
1750 REM * PORCENTAJES *
1760 REM *****
1770 FOR ret=1 TO 2000:NEXT
1780 MODE 1:INK 0,13:BORDER 13:INK
1,0:INK 2,24:INK 3,22:PLDT 0,0,1:DR
AWR 638,0:DRAWR 0,398:DRAWR -638,0:
DRAWR 0,-398
1790 IF na=0 THEN LOCATE 5,7:PRINT
"NO HA ENCONTRADO NINGUNA PALABRA"
ELSE LOCATE 9,7:PRINT "HA ENCONTRAD

```

```

O";na;"PALABRA";:IF na>1 THEN PRINT
"S"
1800 IF nf=0 THEN LOCATE 5,9:PRINT
"NO HA FALLADO EN NINGUNA PALABRA"
ELSE LOCATE 9,9:PRINT "HA FALLADO E
N";nf;"PALABRA";:IF nf>1 THEN PRINT
"S"
1810 LOCATE 7,13:PRINT "PORCENTAJE
DE ACIERTOS:";ROUND(na*100/nj):PR
INT "%"
1820 LOCATE 7,15:PRINT "PORCENTAJE
DE ERRORES:";ROUND(nf*100/nj):PR
INT "%"
1830 WINDOW #7,10,31,21,23:PAPER #7
,1:PEN #7,2:CLS #7:LOCATE #7,2,2:PR
INT #7,CHR$(174);"JUEGA OTRA PARTID
A?"
1840 PLDT 9*16,32,3:DRAWR 22*16,0:D
RAWR 0,48:DRAWR -22*16,0:DRAWR 0,-4
8
1850 RETURN
1860 REM *****
1870 REM * CRONOMETRO *
1880 REM *****
1890 tt=TIME:ti$=RIGHT$(STR$(INT((t
t-tb)/300)),2):IF LEFT$(ti$,1)=" "
THEN ti$="0"+RIGHT$(ti$,1)
1900 LOCATE #2,6,2:PRINT #2,ti$
1910 RETURN
1920 REM *****
1930 REM * FIN DEL TIEMPO *
1940 REM *****
1950 cr1=REMAIN(3):cr2=REMAIN(2):bf
t=1:FOR sd=1 TO 26:BORDER sd:SOUND
1,5*10,10:FOR t=1 TO 100:NEXT:NEXT
:BORDER 13:RETURN
1960 REM *****
*****
1970 REM * COLOCA PALABRA DE IZQUIE
RDA A DERECHA *
1980 REM *****
*****
1990 x=INT(RND*15+1):y=INT(RND*12+1
)
2000 IF x>15 THEN 1990
2010 FOR xx=0 TO 1-1
2020 mp$(x+xx,y)=MID$(p$,xx+1,1)
2030 NEXT
2040 slx=i-1:nx=x+slx:ny=y:sly=1
2050 RETURN
2060 REM *****
*****
2070 REM * COLOCA PALABRA DE DERECH
A A IZQUIERDA *
2080 REM *****
*****
2090 x=INT(RND*15+1):y=INT(RND*12+1
)
2100 IF x<1 THEN 2090
2110 FOR xx=1-1 TO 0 STEP -1

```

## Serie ORO

```

2120 mp$(x-xx,y)=MID$(p$,xx+1,1)
2130 NEXT
2140 slx=-1+1:nx=x+slx:ny=y:sly=1
2150 RETURN
2160 REM *****
*****
2170 REM * COLOCA PALABRA DE ARRIBA
A ABAJO *
2180 REM *****
*****
2190 x=INT(RND*15+1):y=INT(RND*12+1
)
2200 IF y>12 THEN 2190
2210 FOR yy=0 TO 1-1
2220 mp$(x,y+yy)=MID$(p$,yy+1,1)
2230 NEXT
2240 sly=1-1:ny=y+sly:nx=x:six=1
2250 RETURN
2260 REM *****
*****
2270 REM * COLOCA PALABRA DE ABAJO
A ARRIBA *
2280 REM *****
*****
2290 x=INT(RND*15+1):y=INT(RND*12+1
)
2300 IF y<1 THEN 2290
2310 FOR yy=1-1 TO 0 STEP -1
2320 mp$(x,y-yy)=MID$(p$,yy+1,1)
2330 NEXT
2340 sly=-1+1:ny=y+sly:nx=x:six=1
2350 RETURN

```



**AMSTRAD**

Para que tus  
dedos no realicen  
el trabajo duro,  
**AMSTRAD**  
Semanal lo hace  
por ti. Todos los  
listados que  
incluyen este  
logotipo se  
encuentran a tu  
disposición en un  
cassette mensual,  
solicítanoslo.



## ¿Diga...?

—¿Es el **Amstrad** PC/CPC/PCW (táchese lo que no proceda) de ... (a rellenar por el propietario)?

—El mismo, ¿qué quería?

—Verá, soy el ordenador de un amigo de su jefe que me ha pedido que le mande el fichero de datos correspondiente a la última revisión del programa de inventarios.

—¡Ah, sí! Le estábamos esperando.

—Es que llevaba media hora comunicando la línea y hasta el momento no lo había conseguido.

—Ya. Será el jefe que da prioridad a las conversaciones humanas. Bueno, pues manos a la obra, listo para recibir a 300 baudios.

—Listo para transmitir:  
100011101011011... 0100111101110.

—Paquete recibido. Listo para recibir siguiente paquete (protocolo).

—Enseguida:  
011101110101110000111... 1001101100  
1110011. Fin de transmisión  
fichero de datos.

—Recepción concluida.  
Errores en la comunicación=0  
por 100.

—Perfecto. No será necesario repetir la transmisión. Perdone

mi curiosidad camarada **Amstrad**. ¿Qué ha utilizado para modular-demodular la llamada telefónica?

—Un dispositivo modem por acoplamiento acústico de la marca Cat Novation, que ha encontrado mi jefe en Comercial Nuevos Ministerios, en la estación RENFE de la calle Raimundo Fernández Villaverde de Madrid, al precio de 35.000 pesetas. Transmite y recibe en norma standar europea CCITT V-21 (300 baudios) y se conecta al interface RS-232-C. Cuenta con modo *Llamada* y *Respuesta*, seleccionable mediante interruptor externo e indicador de *Comunicación establecida* mediante luz roja. No sobrecarga la alimentación de mis circuitos, ya que dispone de fuente de alimentación independiente.

—Se lo recomendaré a mi jefe. Pues nada más, camarada **Amstrad**, espero que en la próxima ocasión me pase aquel simulador de vuelo del que hablaron nuestro jefes.

—Lo tendré en cuenta. Hasta la próxima.

—Adiós.

## La estrella de la escritura

En impresoras matriciales, la marca Star ha tenido siempre un gran prestigio. Para esta ocasión os presentamos el modelo **Gemini 10X** capaz de escribir a una buena velocidad de 120 caracteres por segundo. Dispone de un buffer de impresión de 816 caracteres, ampliables opcionalmente a 4 K o 8 K. Entre su amplio surtido de tipos de letra podemos destacar la normal (diez caracteres por pulgada), élite (doce cpp), comprimida (diecisiete cop), expandida (cinco cpp), enfatizada, doble picada, itálica etc. juego de 96 caracteres ASCII, otros tantos en itálica, 64 especiales y 32 de gráficos. Se sirve tanto con conector paralelo como serie y es totalmente compatible con los ordenadores CPC de **Amstrad**. Sobre la base



10X también existe una versión con juego de caracteres para PC y otra con carro grande  
—15X— para escribir a 136 caracteres normales por línea.

La encontraréis en Master Computer, Plaza de Cristo Rey, 3, esquina a Cea Bermúdez, 28015 Madrid al precio de 89.000 pesetas.





## Trazando en color

**R**ecomendado para su utilización en el Amstrad PC 1512, ya que dispone de conector paralelo Centronic de 8 bits (los de la serie CPC tienen Centronic de 7 bits) es este

plotter *Sekonic SPL 410* con seis rotuladores de diferentes colores. Permite dibujar en un área efectiva de 385 x 280 mm (papel tamaño DIN A3 o menor) con una velocidad máxima de trazado de 420 mm/seg. El cambio de color se puede realizar a través del software, desde el mismo ordenador, o bien pulsando una tecla en el plotter. Para aplicaciones más técnicas se incluye además un adaptador para utilizar Rotring de tinta china.

Su precio en Peek & Poke, Génova, 11. 28004 Madrid (Tel. 419 81 00) es de 156.800 pesetas.

## No te quedes colgado

Si acabaron los programas de calidad por los programas porque los programas de los terminales cortan de luz. Nos estamos refiriendo, por supuesto, a la fuente de alimentación de seguridad de 700 vatios que podéis ver en la fotografía y que se vende por 78.300 pes. en Micro World, Zurbano, 15, Madrid.

La carga alimentada permite la utilización del ordenador durante el tiempo que necesite, aunque en el que se



podéis ver el fluido el momento en el que se trabaja hasta el momento



## Autodidacta

**P**ara conocer el sistema operativo MS-DOS del Amstrad PC 1512 o bien optas por estudiarte a fondo el libro de instrucciones, o bien te decides por seguir uno de estos cursos en software que editan Logic Control y Turgeon y que podéis encontrar en Master Computer, Plaza de Cristo Rey, 3, 28015 Madrid. Al precio de 5.490 pesetas cada uno.

Ambos cursos, explican Lecciones teóricas y tras cada lección realizan un test práctico para medir los conocimientos adquiridos por el futuro programador.

## Variedad

**A**quí tenéis una buena muestra de la variedad de discos de 5 1/4 pulgadas en doble cara y doble densidad que se ofrecen en Informática Papelería Plaza de Castilla, junto al Asador.

De izquierda de derecha: el disco *Esselte* por 225 pesetas unidad, seguido del *Precision* (220 pesetas), *Verbatim* (225) y *Nashua* (225). Todos ellos también se sirven en cajas de diez unidades.



# A vueltas con los gráficos

Por: Alberto Suñer

*Las imágenes son siempre algo atractivo, mucho más si logramos que nuestras propias creaciones se muevan sin tener que hacer mil operaciones complicadas o definir los gráficos en cada una de sus posiciones.*

**E**n el presente capítulo veremos una rutina que se encargará de realizar esta tarea. Podremos girar un gráfico hasta en cuatro posiciones distintas. De esta forma, si nosotros tenemos una imagen en la posición inicial, al ejecutar nuestra rutina la obtendremos girada en 90°.

Una vez hecho esto, arrancamos de nuevo la rutina y obtendremos sucesivamente el mismo dibujo, pero esta vez con un giro de 180 y 270° con respecto al original.

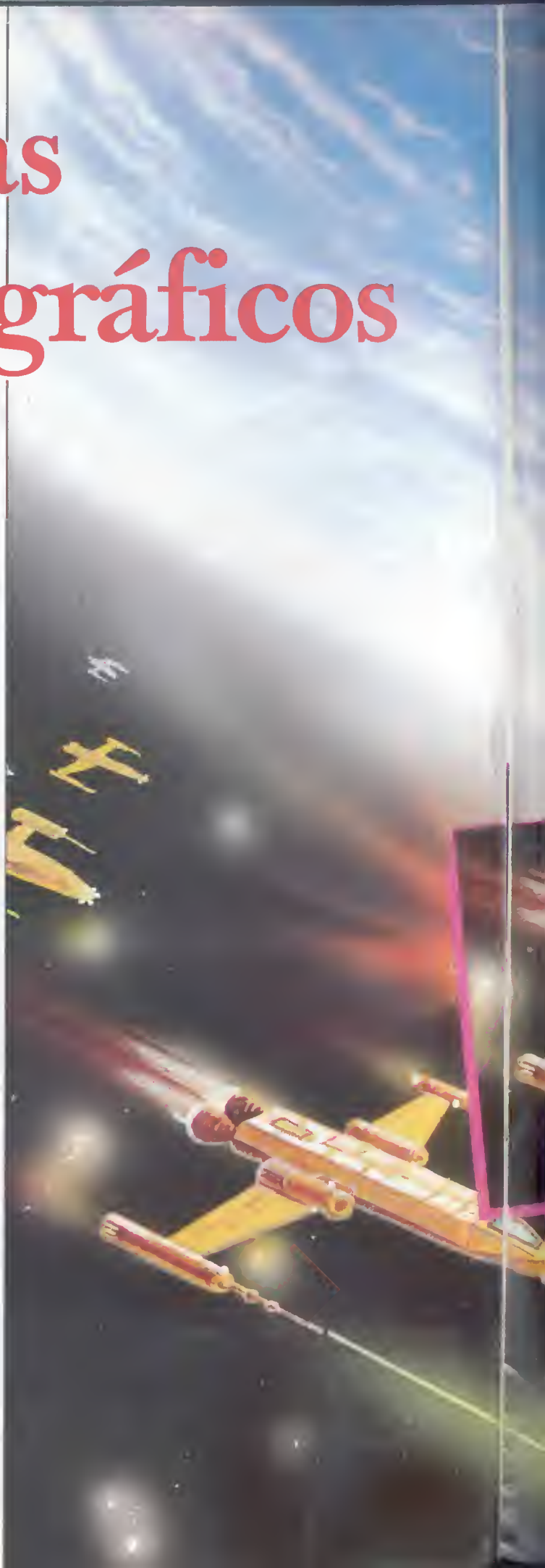
Como siempre que hablamos de gráficos, y dadas las diferentes estructuras de pantallas que posee el **Amstrad**, debemos hacer algunas aclaraciones.

Así pues, nuestra rutina funcionará correctamente sólo con gráficos realizados expresamente para el modo 2 de pantalla, en el cual todos los bits que componen cada uno de los bytes contienen información de pixels encendidos o apagados, sin tener en cuenta el color.

En los gráficos realizados para los demás modos de pantalla, el programa no funcionará, ya que los bytes que componen dicho tipo de gráficos contienen información sobre el color con el que deben ser imprimidos, lo cual dificulta enormemente la operación de giro.

Debemos también apuntar que la rutina está preparada para funcionar con gráficos compuestos de ocho bytes, es decir, con gráficos semejantes a caracteres definidos por el usuario.

Se ha hecho de esta forma para que esta rutina sea utilizable, tanto por aquellos que usan bloques gráficos como para aquellos que prefieren







# Código MÁQUINA

Javier Igual

los gráficos definidos por el usuario.

De cualquier forma, esta rutina puede ser modificada de manera que sea utilizable para bloques de cualquier dimensión, realizando los cambios oportunos en el bucle principal del programa.

## Método de trabajo

Para conseguir el giro de los gráficos en cualquier dirección debemos tomar el bit 7 del primer byte y colocarlo en el bit 7 del octavo (último) byte. El bit 6 del byte número uno debe colocarse en el último bit del byte séptimo, y así hasta llegar al bit uno del primer byte, que deberá colocarse en el bit 7 del mismo byte.

Seguidamente, empezaremos a colocar ordenadamente los bits pertenecientes al segundo

byte de nuestro gráfico. Pondremos el bit 7 de dicho byte en el sexto bit del último byte. El bit 6 del segundo byte, en el bit 6 del byte 7, y así sucesivamente, hasta colocar el bit 0 del segundo byte en el bit 6 del primero.

Después de este trabalenguas, deberemos ejecutar la operación con los ocho bytes que componen nuestro gráfico, colocando cada uno de los bits en las posiciones correspondientes.

Su realización sería como se representa a continuación:

| Gráfico inicial |    |    |    |    |    |    |    | Gráfico final |       |   |  |  |  |  |  |
|-----------------|----|----|----|----|----|----|----|---------------|-------|---|--|--|--|--|--|
| 1               | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 57            | ..... | 1 |  |  |  |  |  |
| .....           |    |    |    |    |    |    |    | .             | ..... | . |  |  |  |  |  |
| .....           |    |    |    |    |    |    |    | .             | ..... | . |  |  |  |  |  |
| .....           |    |    |    |    |    |    |    | .             | ..... | . |  |  |  |  |  |
| 57              | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 64            | ..... | 8 |  |  |  |  |  |

En el gráfico anterior los números del 1 al 64 representan los 64 bits que forman los ocho bytes que componen nuestro gráfico. De esta forma, si, por ejemplo, tomamos el gráfico correspondiente a la letra «A», éste quedaría de la siguiente forma, una vez producido el giro:

|                 |                |
|-----------------|----------------|
| 00011000        | 00000000       |
| 00100100        | 01111100       |
| 01000010        | 00010010       |
| 01000010        | 00010001       |
| 01111110        | 00010001       |
| 01000010        | 00010010       |
| 01000010        | 11111100       |
| 00000000        | 00000000       |
| Gráfico inicial | Gráfico girado |

### Funcionamiento de la rutina

Las dos variables principales que utiliza el programa deben apuntar a dos buffers distintos: uno de ellos debe contener el gráfico inicial y el otro debe reservar un espacio para colocar el gráfico girado.

BUFLET ..... Dirección del gráfico.  
BUFFER ..... Espacio reservado para el gráfico final.

Así pues, debemos cargar el registro doble DE con la dirección donde se encuentra nuestro gráfico (BUFLET) y el registro HL con la dirección del espacio reservado para colocar el gráfico una vez girado.

```
LD DE,BUFLET
LD HL,BUFFER
```

Seguidamente, entramos en un bucle en el cual se irán tomando los ocho bytes que componen la imagen, para entrar más tarde en otro, donde se moverán los ocho bits de cada uno de dichos bytes.

En este último es donde se realiza realmente el trabajo de modificación de nuestro gráfico.

Antes de entrar en el mismo, deberemos cargar en el acumulador el contenido del byte

correspondiente al gráfico, al cual se apunta a través del registro doble DE.

```
LD B,8
LD A,(DE)
```

Seguidamente, rotamos el acumulador hacia la derecha, consiguiendo de esta forma colocar el bit 0 en el *Carry*, para más tarde traspasarlo a la dirección de memoria apuntada por HL (indica el espacio reservado para el gráfico final), mediante una rotación a la izquierda de dicho registro doble.

```
NUD3: RRA
      RL (HL)
      INC HL
      DJNZ NUD3
```

Por último, incrementamos el registro DE y se vuelve a ejecutar el bucle, tomando el siguiente byte correspondiente al gráfico inicial. La rutina, tal como está planteada, lo gira 90° hacia la derecha, con lo cual ejecutándola sucesivamente conseguiremos otra vez el inicial.

Con la misma estructura de nuestro programa,





se podría conseguir efectuar este giro, pero hacia la derecha, consiguiendo girar el gráfico inicial también 90°, pero esta vez en sentido contrario a las agujas del reloj.

Para aquellos que deseen confeccionar una rutina similar, pero para girar bloques de mayores dimensiones, se debe tener en cuenta que, en primer lugar, hay que transformar todos los bytes que componen la primera fila horizontal; una vez hecho esto se deberá pasar a la siguiente línea, hasta llegar a la última.

## Gráfico espejo

A continuación vamos a comentar otro tipo de giro que nos puede resultar interesante a la hora de trabajar con nuestros dibujos.

Se trata de girar un gráfico de forma que se obtenga su imagen reflejada en un espejo. Para un byte cualquiera, quedaría de la siguiente manera:



### PROGRAMA CARGADOR

```
10 FOR N=&A000 TO &A029
20 READ A:SUMA=SUMA+A
30 POKE N,A
40 NEXT
50 IF SUMA<>&A36 THEN PRINT "ERROR EN DATAS"
60 DATA 17,33,160,33,25,160,6
70 DATA 8,197,229,6,8,26,31
80 DATA 203,22,35,16,250,225,19
90 DATA 193,16,240,201,0,0,0
100 DATA 0,0,0,0,0,255,0
110 DATA 0,0,0,0,0,0,0
```

# Código MÁQUINA

0 X X 0 0 0 0 X

Gráfico inicial

X 0 0 0 0 X X 0

Gráfico espejo

El siguiente programa consigue este mismo efecto para un byte:

LD A,(HL)

CPL

LD (HL),A

Suponiendo que el registro doble HL apuntara a la dirección del gráfico, cargaríamos el contenido de dicha dirección en el acumulador, efectuaríamos su complemento y lo cargaríamos en dicha dirección de memoria de nuevo, pero esta vez ya modificado. Así pues, si deseáramos transformar un gráfico compuesto por ocho bytes, deberíamos hacer:

LD HL,GRÁFICO

LD B,8

BUC: LD A,(HL)

CPL

LD (HL),A

INC HL

DJNZ BUC

RET

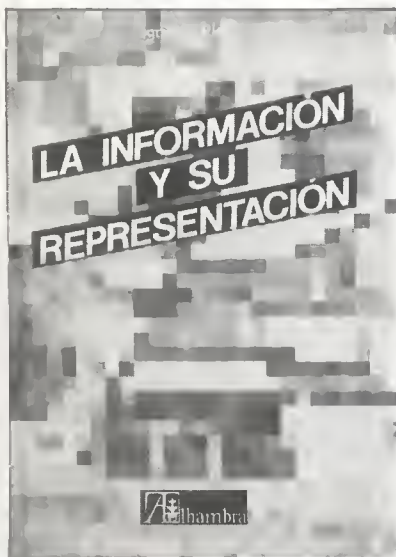
De la misma forma podríamos conseguir la imagen especular de bloques gráficos de mayores dimensiones.

### PROGRAMA ENSAMBLADOR

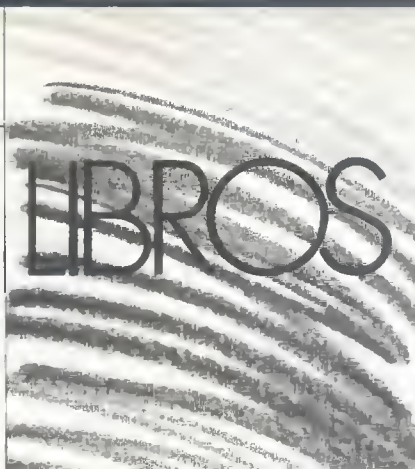
```
10 ORG #A000
20 LD DE,BUFLET
30 LD HL,BUFFER
40 LD B,8
50 NUD4: PUSH BC
60 PUSH HL
70 LD B,8
80 LD A,(DE)
90 NUD3: RRA
100 RL (HL)
110 INC HL
120 DJNZ NUD3
130 POP HL
140 INC DE
150 POP BC
160 DJNZ NUD4
170 RET
180 BUFFER: DEFS 8
190 BUFLET: DEFB 255,0,0,0,0,0,0,0
200 %L+
210 DEFS 0
```

## La información y su representación

Dentro de la serie *Informática en el aula* de la Editorial Alhambra, nos aparece este libro que encaja perfectamente en el marco de la enseñanza de uno de los aspectos teóricos más importantes del ordenador, en el cual descansan todas sus realizaciones prácticas: la teoría de la información. En efecto, sin esto, probablemente no habría ordenadores, aunque tal disciplina se haya enriquecido considerablemente, gracias a las aportaciones de los prácticos. En palabras del propio autor: «El objetivo central de este libro es la teoría de la información, su representación, soporte, estructura y codificación, en el marco de los sistemas de procesamiento de la información, que conocemos con el nombre genérico de ordenadores». Bien, creemos que queda clara, tanto la



función como el tono del libro. Es un libro que va dirigido a técnicos o a educadores, y su lenguaje no es especialmente divulgativo. Hay que reconocer, sin embargo, que la exposición rigurosa y de alto nivel de un tema de este calibre está reñida con todo tipo de licencias artísticas. Creemos que la obra es de gran valor para el maestro, estudiante o especialista del tema, pero tal



vez, para los simplemente curiosos, el autor escale cimas demasiado elevadas.

### Ficha técnica

**Título:** *La información su representación*  
**Autor:** *Ignacio Rierio*  
**Editorial:** *Alhambra*  
**Páginas:** *318*  
**Precio:** *975*

## Programación en Pascal. Turbo Pascal

Si en el mundo de la informática hay un estándar, se llama IBM.

Análogamente, si existe un consenso en los lenguajes de programación para ordenadores de este tipo, se llama Turbo Pascal. Con más de 400.000 copias vendidas en 1985, el Turbo se ha convertido en un producto con el que hay que contar, y a esa realidad responde este libro. Está todo lo que hay que saber acerca del Turbo Pascal. El autor ha tenido el buen sentido de escalonar la profundidad de sus explicaciones, de forma que, tanto el recién llegado al Turbo, como el profesional del compilador puedan extraer un provecho definitivo de sus páginas. Hasta tal punto se ha tenido esto presente en la redacción del libro, que los primeros capítulos podrían llamarse con toda justicia una introducción a las peculiaridades del lenguaje Pascal, así, en abstracto. Por otra parte, todos los programadores saben que

una de las estructuras de datos más comunes y útiles para una aplicación es la gestión de listas encadenadas y la asignación dinámica de memoria.

## Programación en PASCAL TURBO PASCAL

Rodnay Zaks



El autor ha cuidado especialmente el tema, haciendo hincapié en la forma más eficiente de gestionarlas desde el particular punto de vista del Turbo Pascal. Creemos estar ante un buen libro que va a ser útil a todo el mundo.

### Ficha técnica

**Título:** *Programación en Pascal.*  
**Turbo Pascal**  
**Autor:** *Rodnay Zaks*  
**Editorial:** *Anaya Multimedia*  
**Páginas:** *478*  
**Precio:** *3.180*

## NECESITAMOS personas

que conozcan profundamente lenguaje ensamblador del Z80 y dominen los ordenadores Amstrad CPC desde el punto de vista del Lenguaje Máquina. Si usted es uno de ellos póngase en contacto con nosotros rápidamente en la dirección que se indica a continuación, poniendo en lugar muy visible del sobre.

### Referencia

**Código Máquina**  
**AMSTRAD Semanal**  
**Nuestra dirección es:**  
**AMSTRAD Semanal**  
**Carretera de Irún, km 12,400.**  
**Fuencarral. 28049 Madrid.**



# 2 FABULOSOS PROGRAMAS



**¡No te pierdas esta oferta!**  
Envía hoy mismo tu cupón

Oferta válida sólo para España



Ref.-D.1

## RECIBE AMSTRAD Semanal EN CASA

Deseo suscribirme a la revista **AMSTRAD SEMANAL** por un año (50 números), al precio de 7.950 pts. Esta suscripción me da derecho a recibir, totalmente gratis, los dos últimos éxitos de **Dinamic ARMY MOVES** y **GAME OVER** (oferta válida sólo para España).

Nombre \_\_\_\_\_ Fecha de Nacimiento \_\_\_\_\_

Apellidos \_\_\_\_\_

Domicilio \_\_\_\_\_

Localidad \_\_\_\_\_ Provincia \_\_\_\_\_

C. Postal \_\_\_\_\_ Teléfono \_\_\_\_\_

(Para agilizar tu envío, es importante que indiques el código postal)

### Formas de pago

☐ Talón bancario adjunto a nombre de Hobby Press, S.A.

☐ Giro Postal a nombre de Hobby Press, S.A., n.º \_\_\_\_\_

☐ Contra reembolso (supone 125 pts. más de gastos de envío y es válido sólo para España).

☐ Tarjeta de crédito n.º ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Visa ☐ MasterCard ☐ American Express ☐

Fecha de caducidad de la tarjeta \_\_\_\_\_

Nombre del titular (si es distinto) \_\_\_\_\_

(Si pago con la tarjeta de crédito, recibiré un número más de regalo).

Fecha y firma

(Si lo deseas puedes suscribirte por teléfono (91) 734 65 00)

### SOLICITA NUMEROS ATRASADOS, CINTAS SERIE ORO Y TAPAS

☐ Sí, deseo recibir en mi domicilio las cintas que a continuación indico, al precio de 756 ptas. cada una. Cada cinta lleva grabados los programas publicados por **AMSTRAD SEMANAL** durante cuatro números consecutivos (1 al 4, 5 al 8, 9 al 12, etc.)

Las cintas que deseo son:

Números \_\_\_\_\_ al \_\_\_\_\_ Números \_\_\_\_\_ al \_\_\_\_\_  
Números \_\_\_\_\_ al \_\_\_\_\_ Números \_\_\_\_\_ al \_\_\_\_\_

☐ Deseo recibir en mi domicilio los siguientes números atrasados de **AMSTRAD SEMANAL**, al precio de 190 pts. cada uno.

### (Se encuentra agotado el n.º 4)

☐ Deseo recibir en mi domicilio las tapas para conservar **AMSTRAD SEMANAL**, al precio de 850 pts. (No necesita encuadernación).

Nombre \_\_\_\_\_ Fecha de Nacimiento \_\_\_\_\_

Apellidos \_\_\_\_\_

Domicilio \_\_\_\_\_

Localidad \_\_\_\_\_ Provincia \_\_\_\_\_

C. Postal \_\_\_\_\_ Teléfono \_\_\_\_\_

(Para agilizar tu envío, es importante que indiques el código postal)

### Formas de pago

☐ Talón bancario adjunto a nombre de Hobby Press, S.A.

☐ Giro Postal a nombre de Hobby Press, S.A., n.º \_\_\_\_\_

☐ Contra reembolso (supone 125 pts. más de gastos de envío y es válido sólo para España). (Excepto cintas.)

☐ Tarjeta de crédito n.º ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

(Sólo para pedidos superiores a 1.500 pts.)

Visa ☐ MasterCard ☐ American Express ☐

Fecha de caducidad de la tarjeta \_\_\_\_\_

Nombre del titular (si es distinto) \_\_\_\_\_

Fecha y firma

No se admiten solicitudes de cintas contra reembolso



# UULOSOS GRAMAS GRATIS PARA TI

*Suscríbete hoy mismo a  
AMSTRAD y recibirás a  
vuelta de correo los dos  
mayores éxitos de  
Dinamic*



**Benefícate de las ventajas  
de la tarjeta de crédito.  
Un número más, gratis,  
en tu suscripción y la posibilidad  
de realizar el pago aplazado.**



## ARMY MOVES

Como miembro del Cuerpo de Operaciones Especiales, Dendhal ha sido adiestrado en varios sistemas de combate distintos, así como en el manejo de todas las armas, explosivos y técnicas de guerra en la selva. Ahora, tras largos años de entrenamiento, le ha llegado el momento de demostrar sus habilidades y atravesar, por tierra, mar y aire, las líneas enemigas. ¿Lo conseguirá?

## GAME OVER

Los problemas de libertad existen hasta en las más lejanas galaxias. Si no, que se lo digan a los habitantes del planeta Porshaco, quienes están sufriendo en sus carnes viscosas la tiranía de la princesa Gremla. Afortunadamente, Arkos, el más hábil de todos los mega-terminators, ha decidido acabar con este juego; afortunadamente para nosotros, comienza Game Over.

**HOBBY PRESS, S.A.**

Apartado nº 8 F.D.  
28100 ALCOBENDAS (Madrid)

Respuesta Comercial  
Autorización nº 7427  
B.O.C. y T. nº 81  
de 29 de agosto de 1986

No  
necesita  
sello. A  
franquear  
en destino

Respuesta Comercial  
Autorización nº 7427  
B.O.C. y T. nº 81  
de 29 de agosto de 1986

No  
necesita  
sello. A  
franquear  
en destino

**HOBBY PRESS, S.A.**

Apartado nº 8 F.D.  
28100 ALCOBENDAS (Madrid)



# 2 FABULOSOS PROGRAMAS GRATIS PARA TI



## ARMY MOVES

Como miembro del Cuerpo de Operaciones Especiales, Dendhal ha sido adiestrado en varios sistemas de combate distintos, así como en el manejo de todas las armas, explosivos y técnicas de guerra en la selva. Ahora, tras largos años de entrenamiento, le ha llegado el momento de demostrar sus habilidades y atravesar, por tierra, mar y aire, las líneas enemigas. ¿Lo conseguirá?

## GAME OVER

Los problemas de libertad existen hasta en las más lejanas galaxias. Si no, que se lo digan a los habitantes del planeta Porsharo, quienes están sufriendo en sus carnes viscosas la tiranía de la princesa Gremia. Afortunadamente, Arkos, el más hábil de todos los mega-terminators, ha decidido acabar con este juego; afortunadamente para nosotros, comienza Game Over.

*Suscríbete hoy mismo a AMSTRAD y recibirás a vuelta de correo los dos mayores éxitos de Dynamic*

**¡No te pierdas esta oferta!**  
**Envía hoy mismo tu cupón**

Oferta válida sólo para España



**Benefíciate de las ventajas de la tarjeta de crédito. Un número más, gratis, en tu suscripción y la posibilidad de realizar el pago aplazado.**



# POCO RUIDO, MUCHAS NUECES



**1.200 Ptas.** (Versión Cassette)

SPECTRUM 48K  
25-48/170 (1)



**1.750 Ptas.**  
(Versión Cassette)

**ZNI  
CHIP**